DISTRIBUTED SYSTEM SOFTWARE TO CONTROL AN ETHYLENE OXIDE MONITORING SYSTEM

by

Peter Joseph Hawrylak

BS, University of Pittsburgh, 2002

Submitted to the Graduate Faculty of

School of Engineering in partial fulfillment

of the requirements for the degree of

Master of Science

University of Pittsburgh

2004

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This thesis was presented

by

Peter Joseph Hawrylak

It was defended on

June 16, 2004

and approved by

Marlin H. Mickle, Nickolas A. DeCecco Professor, Electrical Engineering Department

Alex K. Jones, Assistant Professor, Electrical Engineering Department

Thesis Advisor: J. Tom Cain, Professor, Electrical Engineering Department

DISTRIBUTED SYSTEM SOFTWARE TO CONTROL AN ETHYLENE OXIDE MONITORING SYSTEM

Peter Joseph Hawrylak, MSEE

University of Pittsburgh, 2004

Government regulations require employers to monitor for harmful chemicals in the work place. Ethylene oxide is one such chemical, but a large number of other chemicals must also be monitored. Existing monitoring systems are targeted at a small range of chemicals or a single chemical. A system to monitor for a new chemical often requires development of a new system. This is expensive and increases the cost of the system. Reuse technology, currently used in numerous areas for rapid product development and for minimizing cost should be applied to the hazardous gas detection field. This thesis describes the development and implementation of software for a distributed system that controls a hazardous gas monitoring system capable of interfacing with multiple types of sensors. The prototype system was targeted at detection of ethylene oxide. Requirements were developed after reviewing existing monitoring systems, and determining desired additional functionality and reuse requirements. Specifications followed from the requirements. Software was designed from the specifications and then implemented and tested.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

The Occupation Safety and Health Administration (OSHA) requires employers to monitor the exposure of employees to harmful chemicals. Employers must monitor instantaneous levels of harmful gases and must report on exposures over a specific period of time. One such gas is ethylene oxide (EtO).

Widely known as a fumigant and as a pesticide, ethylene oxide was discovered to have bactericidal properties in 1929 by H. Schrader and E. Bossert making it ideal for sterilization [1]. Several patents were issued for sterilization processes using EtO between 1937 and 1948 [1]. The Environmental Protection Agency (EPA) registers ethylene oxide as a sterilant [2].

As the development and use of reusable medical instruments continues to increase, so does the need to prevent the spread of disease. Sterilization of medical instruments is critical in preventing infection and the spread of disease. While high temperature sterilization is preferred, medical instruments sensitive to high temperatures require another means of sterilization. Sterilization using ethylene oxide can be performed at low temperatures. Thus, heat sensitive devices, such as those containing plastics and rubber can be sterilized using ethylene oxide. Today ethylene oxide is commonly used as a sterilization agent to sterilize the majority of medical instruments [2].

While ethylene oxide can sterilize equipment, it is also very hazardous to humans. Ethylene oxide has been reported to cause skin, eye, organ damage, and cancer in humans. In addition to health risks, ethylene oxide is extremely flammable and at high concentrations, prone to explosions. The U.S. Department of Health and Human Services has documented numerous cases of explosions at ethylene oxide sterilization facilities. This organization reported one such instance where an explosion of approximately 15 to 20 pounds of ethylene oxide resulted in moving a 50,000 pound sterilization chamber three feet from the foundation [2].

The use of hazardous gas monitoring equipment is critical to employee safety and for the fulfillment of the employer's legal responsibilities. OSHA lists well over 200 air contaminants that must be monitored in the workplace, in Table Z-1 of part 1910.1000. Ethylene oxide is one of the gases on the OSHA list. The National Institute for Occupational Safety and Health (NIOSH) reported that 100,000 health care workers were directly or indirectly exposed to ethylene oxide in 1977 [3].

Significant exposure to EtO results from the release of only a very small amount of ethylene oxide. According to the U.S. Department of Health and Human Services, "1 gram of EtO can initially create a concentration of over 20 parts per million (PPM) in a room … 10 by 10 feet with an 8-foot ceiling" [1]. Many sterilizers use amounts of ethylene oxide greatly in excess of 1 gram. According to OSHA regulation 1910.1047 the amount of ethylene oxide an employee is exposed to must not be above 5-PPM over a 15 minute period, and the 8-hour time-weighted average (TWA) must be below 1-PPM.

Exposures to ethylene oxide can occur for a number of reasons. The most common are as follows. Ethylene oxide is often stored in cylinders with connections to the sterilization chamber. Leaks or failures of the cylinder or connectors can result in the hazardous release of ethylene oxide into the environment [1]. Potential hazardous exposure to ethylene oxide from a leak or failure is not constrained to the sterilization chamber, but can extend to the locations were the ethylene oxide is stored. Accidental release of ethylene oxide from the sterilization chamber can result from a poor ventilation system or from over pressurization of the chamber, which causes the emergency release valves to open [1]. A release of ethylene oxide into the ventilation system can result in a wide spread area of contamination [3]. The door of the sterilization chamber can also develop leaks due to a failure of the door gasket possibly causing a high exposure near the sterilization chamber [1].

The ethylene oxide used in the sterilization chamber must be properly disposed of before the operator enters the sterilization chamber. After the sterilization completes and the ethylene oxide is removed from the sterilization chamber, the ethylene oxide must still be disposed of. One method to accomplish this is by mixing the ethylene oxide with water and releasing the water and ethylene oxide into the sewer system [1]. However, plumbing codes require that an air gap exist between the discharge point and the sewer drain [1]. Without proper control, the air gap, "may be the single most significant routine emission source" [1].

After the sterilization procedure, ethylene oxide can remain in the sterilization chamber. Opening the chamber door in a poorly vented room will result in, "a very short high exposure followed by an increase in the workroom EtO concentration" [1]. Entry into the sterilization chamber to adjust the contents or to clean the chamber can result in an unacceptable exposure [1].

NIOSH, in addition to other precautions, recommends the installation of sensors that respond to elevated levels of ethylene oxide [3]. Further, NIOSH recommends that ethylene oxide concentrations be measured either as an average over a long time, or measured in real-time over a discrete units of time [3]. Continuous monitoring of ethylene oxide concentrations is the best as exposures of short durations can be recorded, and long-term data can be computed from those readings.

Existing detection technologies vary in the ability to react to small concentrations of ethylene oxide and to provide continuous monitoring of short time steps over a long period. Metal oxide sensors are inexpensive but can accurately detect only large amounts of ethylene oxide, such as 20 or 50 PPM [1].

Gas chromatography provides another method of measuring ethylene oxide concentrations. In a gas chromatography system, the ethylene oxide is separated from the sample and then burned causing the sample to ionize [4]. The ionized sample generates a current from which the amount of ethylene oxide can be determined [4]. A disadvantage is that an expert must maintain the gas chromatography system. Training an expert in house or contracting from an third party is expensive [4].

Infrared (IR) sensors can also monitor ethylene oxide. IR sensors measure the absorption of IR by the gas sample to determine the amount of ethylene oxide present in the sample [4]. While capable of monitoring ethylene oxide concentrations below 1 PPM, IR sensors are very expensive and the accuracy is affected by varying levels of moisture [4].

One popular personal air-monitoring device consists of a wearable badge containing chemicals that react to ethylene oxide. The chemicals require removal and testing in a laboratory to determine the exposure. This solution monitors the exposure for the length of time the badge was in use, making detection of small short-term exposures difficult. This method alerts personnel to exposures after the fact, preventing any preventive measures from being taken to reduce exposure [4].

2.0 STATEMENT OF THE PROBLEM

2.1 THE OVERALL PROBLEM

As summarized in the introduction, a number of methods exist to detect and monitor EtO. Each method has strengths and weaknesses. As noted, OHSA and other regulatory requirements mandate monitoring of small concentrations of EtO. Therefore, monitoring systems capable of monitoring small quantities of EtO are necessary. As there are various points of failure in an EtO sterilization system, a complete monitoring system may require multiple monitors to be effective. Both, OSHA and NIOSH recommend monitoring personnel for exposure to EtO over the long term and for short term exposures [2]. A system capable of monitoring, in real-time, exposures over the long term, and also capable of providing indications of short-term exposures would provide data for both NIOSH recommended monitoring methods. Providing the current amount of EtO present allows personnel to take precautions to avoid unnecessary exposure to EtO.

Gas detection systems currently in production often require the replacement of the entire system, rather than just replacement of the sensor when a sensor is no longer operating properly. The need to replace the majority of the system when only the sensor needs replaced can be expensive. Allowing for replacement of only the sensor would reduce operating cost.

Several systems offer sensors capable of detecting several different gases. These systems can suffer from inaccurate readings due to cross contamination from other gases as the sensor may be sensitive to gases not being monitored.

Development of a monitoring system specifically for each gas would increase the development and purchase cost of a monitoring system. One way to reduce the cost of the system is the identification of components that can be reused. A reusable base system containing the detection, alarm, safety, and maintenance functionality can reduce the overall cost. Once the base system is developed, replacement of the entire system would not be required when interchanging sensors.

To provide the plug and play ability described above, the base system must be able to interact with sensors designed to detect specific gases in the environment. The sensor must be easily replaceable and the system must be able to accept sensors of different types and sensors for detection of different gasses. The type of sensor currently installed in the system will determine the gas(es) the system will monitor. Extension of the system to monitor other gases would require only the development of a sensor sensitive to the desired target gas.

Providing warning of hazardous conditions to personnel is the primary purpose of the monitoring system. The NIOSH recommends that, "audible and visual alarms should be activated by [EtO sensors]" [5]. A visual indication of the current PPM allows personnel to be aware of the current PPM, and allows personnel to respond when the PPM increases. An audible alarm system with multiple settings would alert personnel who cannot see the PPM displayed.

The PPM display must also be easily viewable by personnel. Placing the sensor some distance away from the monitoring system would allow the monitor to be placed in an easily accessible and viewable area, such as a hallway or central station, while the sensor is monitoring an area that is difficult to access, such as a ventilation duct or chemical storage room. The system would then be able to show the current concentration of EtO in a central location. The system must allow the portion of the system containing the gas sensor to be some distance from the display portion of the system.

The system must compute and display the PPM to the users and an interface must be provided for technicians to perform normal maintenance operations on the system. Warning personnel of unsafe conditions can prevent accidental exposures. It is therefore critical for a monitoring system to provide support to display information about one location in several other locations. This allows personnel, not yet exposed to avoid a dangerous area. The system must provide a local display and built-in user interface for normal maintenance operations. Providing multiple displays of the PPM is beneficial. For example, a sterilization room with multiple entrances should have a display indicating the PPM at each entrance. Another example is displaying the PPM in a hallway located by a sterilization room, at all entry points into that hallway. Providing multiple displays allows personnel to be aware of and possibly avoid a dangerous exposure. The system must be capable of forwarding the display information to additional remote display repeaters.

The ability to control and to monitor a number of systems from a central location would greatly increase the effectiveness of the entire monitoring system. Centralized monitoring of several sensors enables one person to monitor quickly and efficiently a large area containing multiple monitoring systems. Identification of unsafe conditions would be quick and additional actions can be taken to ensure safety. An external remote system should be responsible for monitoring the operation of several different systems and for computing and storing long-term data. In addition to monitoring, the ability to perform normal maintenance on a monitoring system remotely would allow basic maintenance to be performed quickly and easily on multiple systems. The system must provide the ability to connect to and be controlled from this remote system.

As previously described, any specialized training required to maintain existing systems contributes to the high cost of a monitoring system. One way to reduce the overall cost is to reduce the maintenance cost of the system. Providing easy to use functions for common maintenance and automating portions of maintenance functions where possible will reduce the amount of training required to maintain the system. Ideally, an in-house technician with training and a system manual should be able to perform basic maintenance operations. The system must provide an easily understandable man-machine interface for PPM display and maintenance purposes, and the ability to be controlled from a remote computer for long-term monitoring and maintenance purposes.

The system requirements discussed above for a replaceable sensor and the ability to separate the sensor and the primary display require the development of a distributed system. To accomplish this, the system must be broken up into several interacting components. The base system, as shown in Figure 2-1 below, was partitioned into three separate entities, the Sensor Module, the Sensor Unit, and the Area Monitor. Each of the three components is responsible for a different part of the system functionality. The remote computer and the Remote Display components shown in Figure 2-1 are not part of the system in development, but are required to be developed in the future. The remote computer enables the user to remotely control the system, and the remote display will repeat the information shown on the primary system display. The system is designed to interact with these two additional components, using a custom communications protocol.

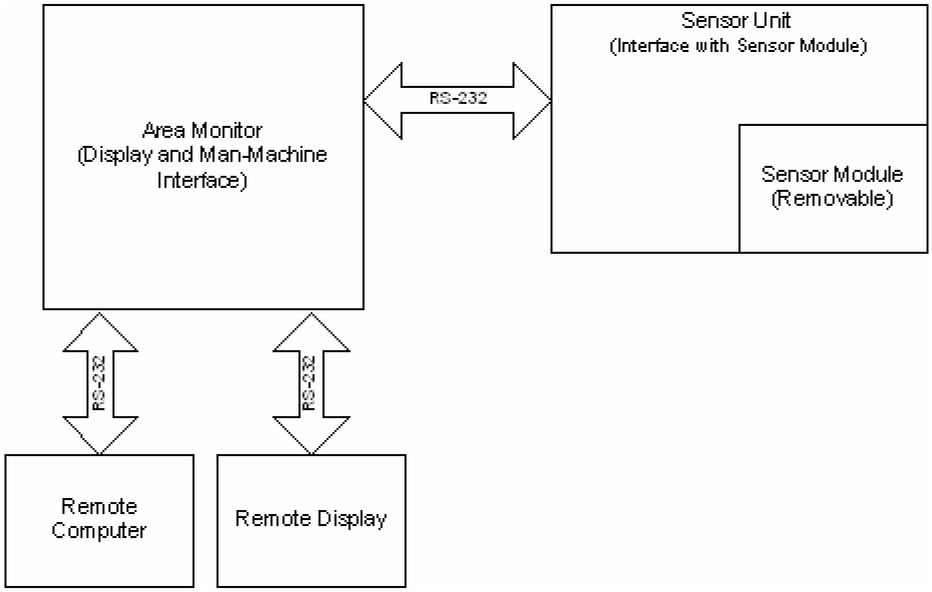

Figure 2-1: Top-Level System Block Diagram

The Sensor Module contains the chemical sensor and data associated with the particular sensor. This data consists of operational data, which the system must have to accurately interpret the chemical sensor output. The chemical sensor, surrounding hardware, and parameters specific to that chemical sensor must be able to be easily replaced. To allow for easy replacement of sensors, the Sensor Module must be a separate component in the system.

As the Sensor Unit containing the Sensor Module is required to be capable of being located up to 25 feet from the display, the Sensor Unit must be separate from the display and man-machine interface unit. The Sensor Unit must interface with the Sensor Module to perform the calculations for the various tasks (i.e. PPM calculations, calibration). The Sensor Unit must detect the removal and the insertion of the Sensor Module, and must forward this information onto the remaining components in the system. As local processing must occur at the Sensor

Unit, there must be a processor in the Sensor Unit. The system must detect the removal of the Sensor Module, and detect the insertion of the new Sensor Module.

The system must inform the user of the current PPM and system state, as well as allow the user to perform maintenance operations. Hence, the system must contain a display, and a built-in man-machine interface. The Area Monitor contains the primary system display and man-machine interfaces. The Area Monitor contains the processor responsible for controlling the man-machine interfaces (remote PC and keypad) in the system and for controlling the user operation of the system. All user requests and information from the user must go through the Area Monitor. The system must be capable of being controlled remotely and must provide the ability for a display repeater to be attached to the system. The Area Monitor must provide communication links between the Sensor Unit and itself, between a remote computer and itself, and between a Remote Display and itself.

2.2 STATEMENT OF THE PROBLEM

A system capable of monitoring small amounts of EtO and capable of warning personnel and taking action to prevent further exposure is greatly needed. Such a system must accurately monitor the PPM of a specific gas, inform the user of the current concentration, and warn personnel if unsafe levels are detected. Providing connections for remote monitoring and control systems provide increased safety as large areas can be monitored in a central location. Low cost allows multiple systems to be deployed to provide comprehensive coverage.

As described above, multiple monitoring systems may be needed to provide adequate monitoring and protection of personnel. The cost of the system must not be prohibitive to the purchase of the multiple systems required to provide adequate protection. A reusable base system capable of interfacing with multiple types of sensors reduces the cost of monitoring systems. System maintenance operations cannot require extensive training to perform. Monitoring capabilities are enhanced by allowing the chemical sensor to be separated from the primary display, and by providing interfaces to a remote computer and to remote displays.

The system architecture shown in Figure 2-1 is the result of analyzing the top-level system requirements. The system architecture is that of a distributed multiprocessor system. The problem to be addressed in this thesis is the development of the software system to control the overall system.

The system requirements and specifications are developed in Section 3. The system design is presented in Section 4. Testing and implementation of the software is described in Section 5, and conclusions and contributions to the field are presented in Section 6. I would like to thank CCS Inc. for providing the PCH compiler to compile the C code used in the PIC processors.

3.0 REQUIREMENTS AND SPECIFICATIONS

3.1 FUNCTIONAL REQUIREMENTS AND SPECIFICATIONS

The primary purpose of the system is to alert personnel to the presence of harmful levels of a specific gas in the environment. This specific system must be able to detect the current concentration of EtO in the environment. The system must provide a means of displaying the current concentration of EtO, so that personnel are aware of the amount of EtO in the environment. The system must provide a warning to personnel of unsafe concentrations. The system must visually and audibly indicate the presence of unsafe operating conditions.

The chemical sensors used in detection of EtO have a limited shelf life and operational life, after which the sensor's ability to monitor EtO degrades. Sensors specific to other gases have a similar limited shelf life and operational life. The system must prevent an expired sensor from being used and detect when a sensor currently in operation expires.

The system must operate with a variety of sensors, capable of detecting specific gases. Replacement of the Sensor Module must be easy. The sensor must be connected to the system for the PPM to be computed. The system must a provide means to determine when the chemical sensor is removed and replaced. The system must detect the absence of a sensor and report that fact to the user.

The EtO sensor used in development of the base system is an electrochemical sensor. The sensor utilizes a chemical reaction that is influenced by the presence of EtO. The chemical reaction produces a current, which can be measured to determine the amount of EtO present in the environment. The electrochemical reaction in the sensor is not stable and requires a variable amount of time to stabilize to a new environment. When the sensor is placed in the unit or the unit is powered up, the system must determine when the sensor is ready for normal operation after replacement.

As previously stated the output of electrochemical sensors can vary over time. The system must compensate for this variation to maintain accuracy. The majority of sensors require recalibration during operation. The calculated PPM begins to loose accuracy when the sensor output at the calibration points differs from the stored calibration points. The system must provide calibration functions to account for the variation of the electrochemical sensor. The calibration must determine the calibration point automatically from the chemical sensor output.

The system must be capable of monitoring different gases, and must handle a wide range of alarm thresholds. For example, OSHA requires employers to monitor personnel for concentrations around 5 PPM for EtO, while for carbon monoxide concentrations around 50 PPM must be monitored [2][6]. In addition to alarm thresholds, the time required for a valid calibration also depends on the gas. The system must provide for the alteration of alarm thresholds and the calibration durations for proper operation when switching between target gases.

A number of faults can develop during operation that would impair the ability of the system to function correctly. The system must be capable of monitoring itself for these faults. The functionality of the system must be periodically checked to ensure the operability of the system.

Therefore, the system must perform a periodic diagnostic to test functionality. The user must also be able to initiate a diagnostic.

The purpose of the monitoring system and the installation site determine some of the system operational parameters. While these settings should rarely change, the system must be able to adapt to new requirements and regulations. The system must provide for the initial factory programming of the settings into the system, and must provide for these settings to be updated.

The system must support a connection to a remote computer. The remote computer provides for centralized monitoring and control of multiple systems. To provide these monitoring and control functions the remote computer must be able to collect the operational data used in the system and to query the current system status.

The user must be able to perform system operations to maintain the system during usage. The system must provide the user with two man-machine interfaces. The primary interface must be built into the system, while the second interface must provide a protocol for a remote computer to interact with and control the system. The system must provide the ability to display available options and data to the user. The built in man-machine interface must allow the user to perform system calibrations, update the system configuration, initiate a system diagnostic, display any outstanding faults, and allow the user the ability to initiate action to clear the outstanding fault. The remote computer must provide the user with the ability to perform all operations available through the built in interface along with additional configuration and information gathering functions. For the purposes of allowing the PPM to be displayed in several locations, the system must also provide a connection to an external display repeater. The display repeater must be able to be located up to 25 feet from the system display.

The integrity of the stored system operation data can be disrupted if the system looses power. Therefore, the system must detect when a power loss may have affected the system operation data and must inform the user of this occurrence.

3.1.1 Calculation of PPM Requirements and Specifications

The PPM calculated by the system represents the amount of gas detected in the environment. Detecting, elevated PPM levels indicates a harmful amount of gas in the environment, allowing workers to take necessary precautions and the activation of control devices. Exposure levels specified by regulatory groups such as OSHA, and the EPA are stated as PPM over a specific interval of time. The system must compute the current PPM value periodically to provide the current PPM and data for the regulatory requirements.

The system must compute, display, and report the PPM at regular intervals. Figure 3-1 below shows the use case diagram for the calculation of the PPM. A timer triggering the calculation of the new PPM at the specified frequency must initiate the calculation. As shown, the timer will inform the Area Monitor that it is time for the new PPM calculation. The Area Monitor, in response to the timer, will request the current PPM to be calculated by the Sensor Unit. The PPM value calculated from the sensor output must be compensated for the temperature of the environment and the altitude of the system. The Sensor Unit must sample the Sensor Module output and perform the mathematical calculations to compute the PPM. The system must also obtain the temperature of the surrounding environment. After the PPM is calculated and compensated, the Sensor Unit must send the PPM to the Area Monitor. Upon receiving the current PPM, the Area Monitor must display the PPM and operate the alarms and relays as needed. The Area Monitor must forward the PPM reading to the remote computer and to the Remote Displays.

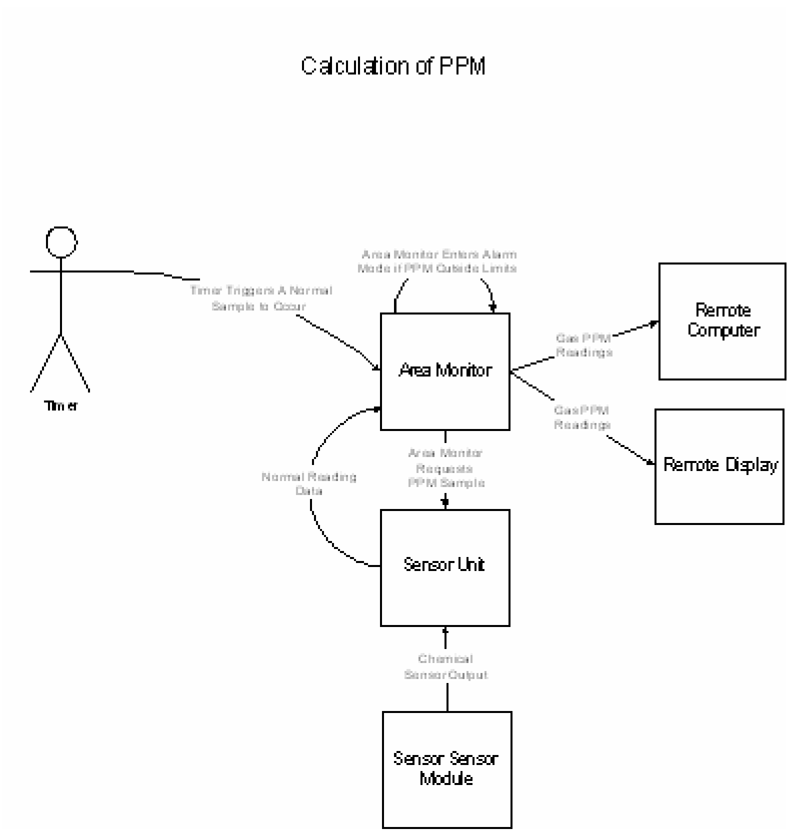

Figure 3-1: Use Case Diagram Showing the Calculation of the PPM

The specifications for the calculation of the PPM for the Sensor Unit are derived from the requirements discussed above. The Sensor Unit is responsible for collecting the data needed, performing the computations to obtain the PPM and sending the PPM to the Area Monitor when requested.

The output of the chemical sensor is influenced by temperature and altitude, in addition to the current amount of EtO present. To calculate the current PPM the system will obtain the output of the chemical sensor, the current temperature, and the current altitude of the system. Hence, the system will display the equivalent PPM detected at the baseline temperature and altitude. The sequence diagram shown in Figure 3-2 shows the sequence of events for the calculation of a new PPM sample.

The Sensor Unit will receive the request PPM message from the Area Monitor, indicating that a new sample is to be taken. The Sensor Unit will sample the chemical sensor output, and will obtain the current temperature.

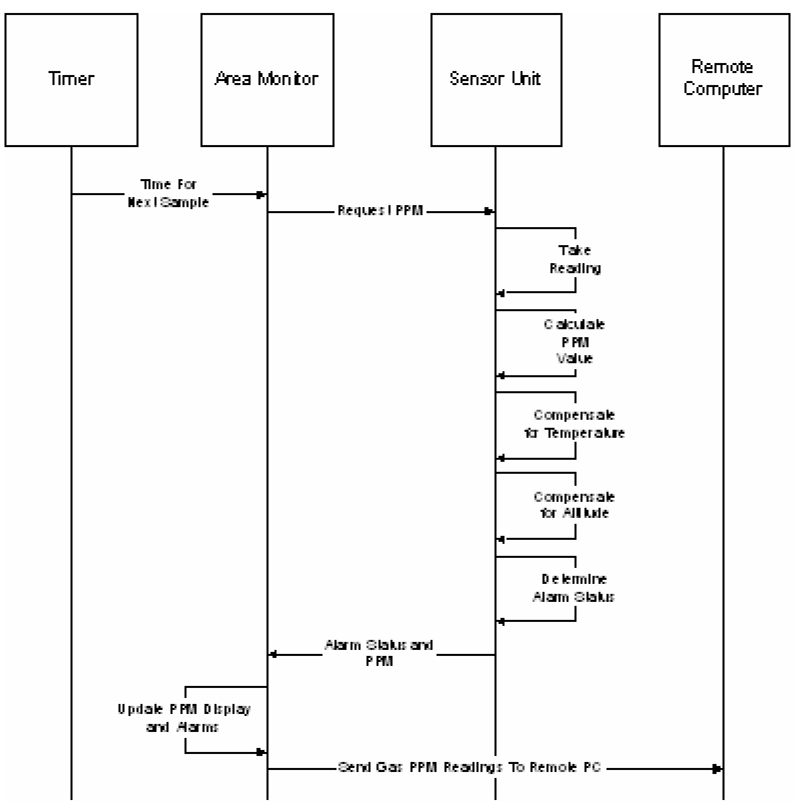

Figure 3-2: Sequence diagram for calculation of PPM.

The method used to compute PPM from the chemical sensor output is described below. Three compensations are required in the calculation of the PPM. The chemical sensor output is composed of two parts. The first part of the output is a background current produced by the chemical sensor, based on the current temperature. The background current is unrelated to the current PPM level and must be removed from the reading before the calculation of the PPM. The second part of the chemical sensor output is the reading proportional to the current PPM of EtO. The part of the reading proportional to the PPM must be used to calculate the PPM.

The sensor output proportional to the current PPM of EtO is further influenced by temperature and altitude. In both cases, the output is inflated or deflated from a baseline temperature and altitude. The system must operate in a wide range of temperatures and altitudes and the PPM readings must be converted to PPM at the same baseline temperature and altitude conditions.

The sensor manufacturer must provide data points describing the effect of temperature and altitude on the sensor output for the three compensations previously described. The Sensor Unit must be capable of performing the compensations based on the current temperature and altitude and from the data points. Piecewise linear approximations for all three compensations fitted to the manufacturer provided data points will be used to determine the compensation values during the calculation. The piecewise linear approximations allow the compensation multiplier to be calculated accurately for any temperature and altitude within the system range.

The first compensation, denoted as f1, removes the background current from the reading. The second compensation, denoted as f2, adjusts the output to the calibration temperature. The third compensation, denoted as f3, adjusts the output to the baseline altitude. This leads to the development of three functions used to obtain the three compensation factors, where f1 and f2 are based on temperature, and f3 is based on altitude.

The calibration data obtained from a calibration will be used to interpret the current reading. The output of the chemical sensor in a 0-PPM environment, denoted as ADC_ZERO_CAL, the temperature at which the calibration was performed, denoted as $\text{Temperature}_{\text{calibration}}$, and the factor to convert the sensor reading into PPM units, denoted as SLOPE, will be used in the calculation.

The Sensor Unit first samples the chemical sensor output obtaining the current reading, denoted as ADC_VAL. The output of the chemical sensor at 0-PPM and the background current must be removed from ADC_VAL. Equation 1 shows the computation of ADC_ZERO_COMP, is the value used to remove the background current and the zero offset from ADC_VAL.

$$\text{ADC\_ZERO\_COMP} = \text{ADC\_ZERO\_CAL} + \begin{pmatrix} \text{f1}(\text{Temperature}_{\text{current}}) \text{ -} \\ \text{f1}(\text{Temperature}_{\text{calibration}}) \end{pmatrix} \times \left(\frac{1}{SLOPE} \right)$$

Equation 1

Next, the raw PPM is calculated, denoted as PPM_{RAW}. The PPM_{RAW} is obtained as shown in Equation 2. The ADC_ZERO_COMP, previously calculated, is subtracted from ADC_VAL, leaving only the chemical sensor output proportional to the current PPM of EtO present. The SLOPE is then used to convert the resultant sensor reading to units of PPM.

$$PPM_{RAW} = SLOPE \times \left(ADC\_VAL - ADC\_ZERO\_COMP \right)$$

Equation 2

Finally, the raw PPM value must be converted to an equivalent PPM at the altitude and temperature of the latest calibration. Equation 3 illustrates this process.

$$PPM_{COMP} = \frac{\left(PPM_{RAW} \times \left(\frac{f2\left(Temperature_{calibration}\right)}{f2\left(Temperature_{current}\right)} \right) \right)}{f3\left(Altitude\right)}$$

Equation 3

The Sensor Unit must determine the alarm status based on the newly calculated PPM value. The system must support four alarm statuses, no alarm, low alarm, high alarm, and over-range. The no alarm status must indicate that the current PPM is below the low and high alarm thresholds. The low alarm status must indicate a current PPM above the low alarm threshold, but below the high alarm threshold. The high alarm status must indicate the current PPM is above both the low and high alarm thresholds, but within the sensor range. The over-range status must indicate that the current PPM is above the maximum PPM output of the sensor. The Sensor Unit will send the alarm status, followed by the PPM value to the Area Monitor.

The specifications for the calculation of the PPM for the Area Monitor are derived from the previous requirements. The Area Monitor must update the PPM display periodically and control the alarms. The Area Monitor will update the display and alarms to reflect the new PPM, upon receiving the status and PPM from the Sensor Unit.

The Area Monitor will initiate the PPM calculation process every second by requesting the PPM from the Sensor Unit. The Area Monitor will wait for the Sensor Unit to send new alarm status and PPM. Based on the status information sent with the PPM, the Area Monitor must determine what actions to take with respect to the alarms and relays. The Area Monitor must then send the PPM and status information to the remote computer and Remote Display.

The Area Monitor will activate the alarms and alter the display based on the alarm status. The Area Monitor must indicate visually and audibly the current alarm state. The Area Monitor must use varying colors and flashing of the PPM display to indicate, visually, the current alarm status. The system will use three different display colors to help visually indicate the alarm status. Green will indicate no alarm status, yellow will indicate low alarm status, and red will indicate high alarm or over-range status. The system will flash the display on and off when the current alarm status is low, high, or over-range. The display will not flash when the current alarm status is no alarm. The system will display an over-range condition by displaying "OVER" on the display, not the PPM. The Area Monitor will activate the alarm horns when the alarm status is low, high or over-range. The Area Monitor must also be capable of activating up to two external controls in response to the detection of elevated levels of gas. The Area Monitor will activate these relays when the alarm status is low, high, or over-range.

3.1.2 Calibration requirements and specifications

As indicated previously, chemical sensor output is based on an electrochemical chemical reaction. This chemical reaction is sensitive to EtO and the sensor generates output, which can be correlated into PPM of EtO present. The chemical reaction is unstable and the operating state of this reaction can change over time and altering the sensor output.

The drift in the sensor output described above requires recalibration of the system. Recalibration will correct the error introduced into the PPM readings from the sensor drift. Sometimes the drift is confined to the background current, described in Section 3.1.1, requiring recalibration of only the zero point. The user must be able to calibrate the system to maintain the required accuracy.

3.1.2.1 Span and Zero Calibration

The manufacturer must perform the initial calibration of the Sensor Module at manufacture time, and the user in the field may need to perform a full calibration periodically. The Span and Zero Calibration is a full calibration of the system. The calibration must compute the factors used to convert the sensor output into units of PPM. To obtain these conversion factors two data points must be obtained from the sensor. The system will use the sensor reading at 0-PPM, known as the zero point, and the sensor reading at a known PPM, denoted as the reference point. The Span and Zero Calibration will find and use these two points to calculate the conversion factors.

As Figure 3-3 shows, the remote or local user initiates this operation and must be able to enter the required data for the reference calibration. The system must then obtain the calibration points and determine the new conversion factors. The system requires information about the reference gas used for the Span Calibration. The system must provide the user the ability to enter information describing characteristics of the reference gas. From this information, the parameters used to convert the Sensor Module output into PPM must be updated. Often a less toxic gas is used in place of the toxic gas the sensor is designed to detect, for the span calibration. For safety reasons, the system must allow the Span Calibration to be performed with gases other than EtO, such as Carbon Monoxide (CO). To ensure a valid calibration has taken place, the system must compare the resultant values to limits specified by the sensor manufacturer, and must ensure that both points were found within the respective time limits. The system must inform the user when the calibration has completed.

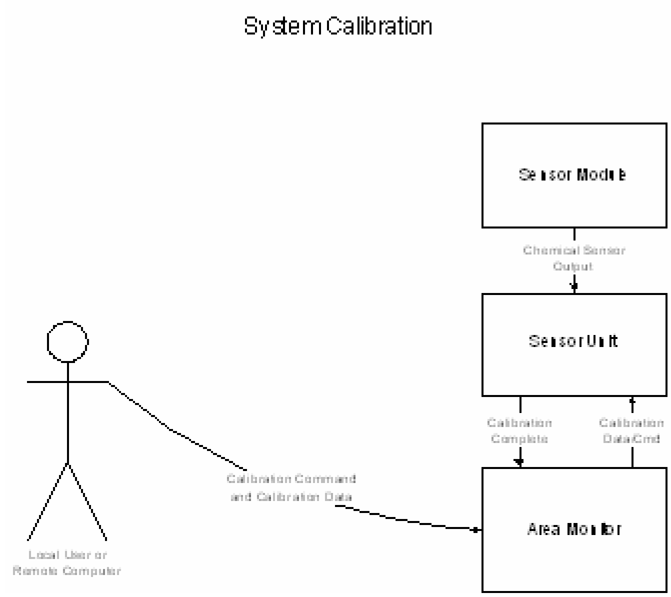

Figure 3-3: Use Case Diagram Showing the user performing a calibration.

From the Span and Zero Calibration requirements previously defined, specifications are developed for the Span and Zero Calibration routine for the Sensor Unit. Figure 3-4 shows the sequence of events for a Span and Zero Calibration. The Sensor Unit will begin a Zero and Span Calibration upon receipt of the zero and span calibration command from the Area Monitor. The Sensor Unit will then obtain the zero point, monitoring the sensor output for stability. The Sensor Unit will determine stability comparing the slope of the least squares fit to 10 samples to the manufacturer specified maximum slope value. The reading is stable when the slope is less than the maximum slope.

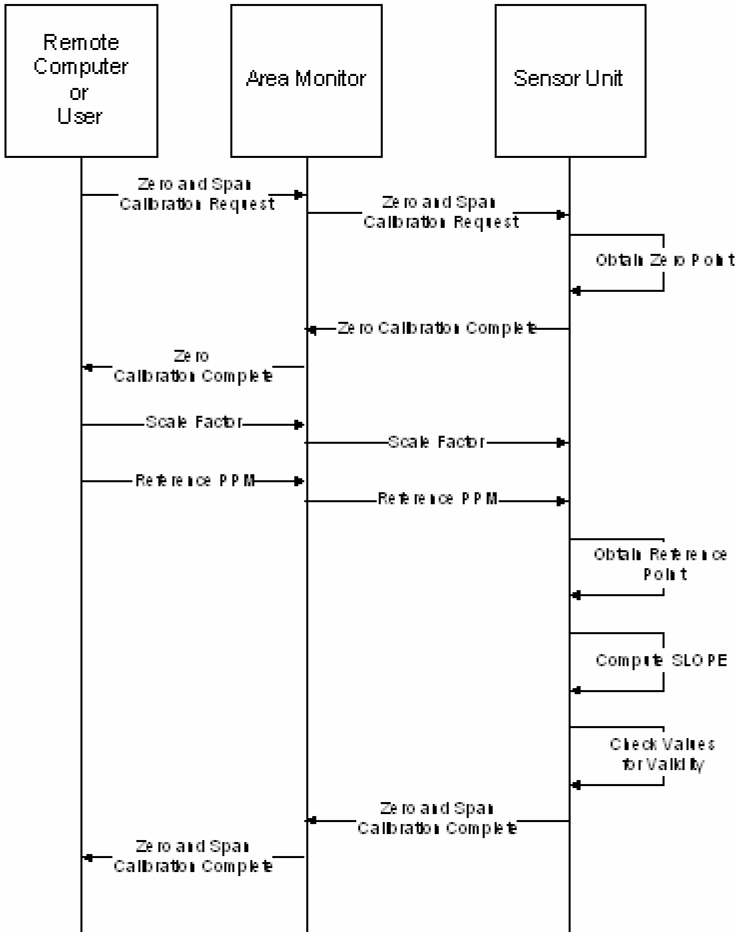

Figure 3-4: Sequence diagram for the Zero and Span Calibration

The Sensor Unit will send the zero calibration complete message to the Area Monitor, and will wait to receive the scale factor and the reference PPM from the Area Monitor. After receiving the scale factor and the reference PPM the Sensor Unit must obtain the reference point, in a manner similar to that of the zero point. The Sensor Unit will obtain the reference point in a similar manor to the collection of the zero point. The conversion from sensor output to units of PPM requires the slope of the line connecting the zero point and span point to be calculated. The slope will be computed from the zero point and span point calculated during the span and zero calibration, as shown below in Equation 4. The PPM of the reference gas is denoted as PPM_CAL, the sensor output at 0-PPM is denoted by $ADC\_ZERO\_CAL_{S\&Z}$, and the sensor output at PPM_CAL is denoted as $ADC\_CAL_{S\&Z}$.

$$SLOPE = \frac{(PPM\_CAL - 0)}{(ADC\_CAL_{S\&Z} - ADC\_ZERO\_CAL_{S\&Z})}$$

Equation 4

The new conversion values obtained from the calibration will be checked for validity against the manufacturer's parameters. The SLOPE must be within a valid range specified by manufacturer parameters. The Sensor Unit will store the new calibration values and will inform the Area Monitor that the Span and Zero Calibration has completed.

The specifications for the Area Monitor for the Span and Zero Calibration are derived from the requirements stated above. As shown in Figure 3-4, the user or remote computer must initiate the span and zero calibration. The Area Monitor will send the span calibration command to the Sensor Unit, in response to user input. The Area Monitor will wait for the Sensor Unit to send the zero calibration complete message, indicating the zero point was found. The user must enter the scale factor and the PPM of the reference gas. The Area Monitor will then prompt the user to enter the data required for the reference calibration. The Area Monitor must check the reference PPM entered by the user if the Span and Zero Calibration is performed using the built-in interface. The reference PPM will be no greater than the maximum sensor range divided by the scale factor as a percentage. The Area Monitor will convert any floating-point numbers to Microchip floating-point format. The Area Monitor will send the scale factor first and reference PPM second to the Sensor Unit. The Area Monitor will wait for the Sensor Unit to send the zero and span calibration complete message, indicating that the new conversion values have been obtained. The Area Monitor will send the zero and span calibration complete message to the remote computer.

3.1.2.2 Zero Calibration

The output of the chemical sensor in a 0-PPM environment can vary greatly in the field because of the instability of the reaction. The technician must therefore recalibrate the system to correct for the change in the sensor output to a 0-PPM environment by performing a Zero Calibration. The Zero Calibration is the most common calibration operation that is performed in the field.

The Zero Calibration function must allow the user to have the system find a new zero point. This point will correspond to the sensor output in a 0-PPM environment. From this information, the parameters used to convert the Sensor Module input into PPM must be recalculated to reflect the new zero point. The system must automatically determine when the sensor output reading has stabilized to the 0-PPM environment. The system must check to ensure that the resulting data is within the manufacturer's limits and that the zero point was found within the allotted time. The user must initiate a zero calibration. The Sensor Unit must inform the Area Monitor and the Area Monitor must inform the user when the Zero Calibration has completed.

The specifications for the Sensor Unit for Zero Calibration follow from the previously described requirements. Figure 3-5 illustrates the sequence of events during a Zero Calibration. The Sensor Unit will receive the zero calibration start command, and will then begin to monitor the sensor output for stability. The Sensor Unit will determine when the 0-PPM point has been reached. The slope of the least squares fit line of 10 data points will be used to determine stability of the output. The reading is stable when the slope is less than the manufacturer provided maximum slope.

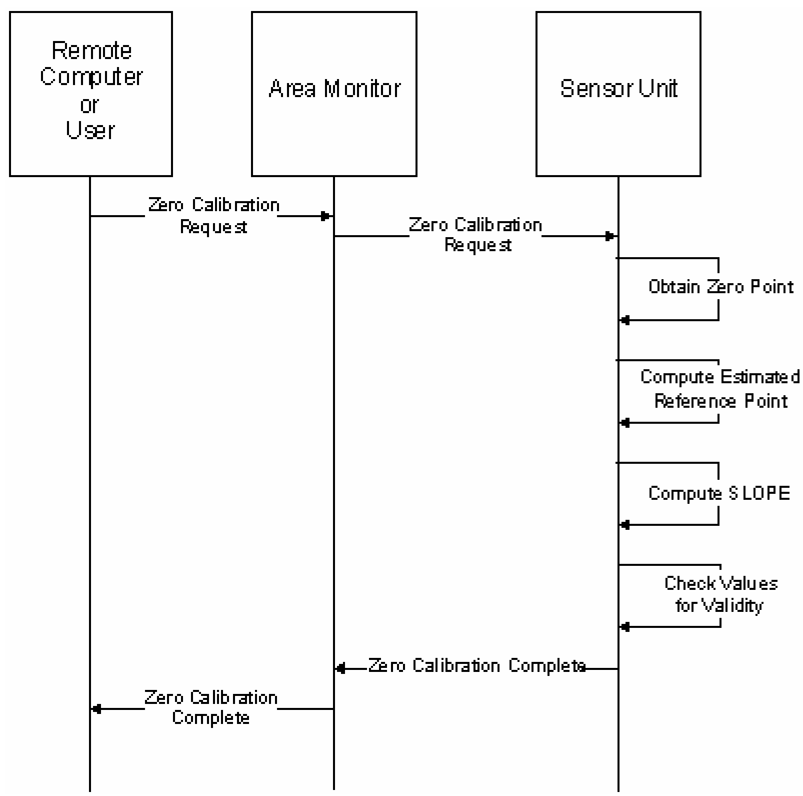

Figure 3-5: Sequence diagram of Zero Calibration..

The conversion from the sensor output to PPM requires the slope of the line connecting the zero point and span point to be calculated. The zero calibration produces only the zero point, so an extrapolated reference calibration point will be determined using the zero point just found, the old reference point, and the current temperature. The use of the extrapolated reference calibration point will improve the accuracy of the calculated conversion values.

The reference calibration point is derived as follows during a Zero Calibration. The Zero Calibration obtains the output of the chemical sensor in a 0-PPM environment, denoted as $ADC\_ZERO\_CAL_{Z}$ for a Zero Calibration and as $ADC\_ZERO\_CAL_{S\&Z}$ for a Span and Zero Calibration. The sensor output at a known PPM will be denoted as ADC_CAL_{Z} for a Zero Calibration and as $ADC\_CAL_{S\&Z}$ for a Span and Zero Calibration.

The method to calculate the compensated PPM given in Equation 3 can be rewritten as shown in Equation 5.

$$PPM_{COMP} = \left[\left(ADC\_CAL_{S\&Z} - ADC\_ZERO\_CAL_{S\&Z} \right) \times \left(\frac{f2\left(Temperature_{S\&Z}\right)}{f2\left(Temperature_{Z}\right)} \right) \right] \times SLOPE \times f3\left(Altitude\right)$$

Equation 5

The bracketed quantity is in units of sensor output units. The bracketed quantity is the $ADC\_CAL_{S\&Z}$ value adjusted to the temperature at which the current Zero Calibration is performed and has the background effect of the last calibration removed. The $ADC\_ZERO\_CAL_{Z}$ can then be added to the bracketed quantity to reflect the current background effect. Equation 6 shows the computation of ADC_CAL_{Z}.

$$ADC\_CAL_{Z} = ADC\_ZERO\_CAL_{Z} + \left[\left(ADC\_CAL_{S\&Z} - ADC\_ZERO\_CAL_{S\&Z} \right) \times \left(\frac{f2\left(Temperature_{S\&Z}\right)}{f2\left(Temperature_{Z}\right)} \right) \right]$$

Equation 6

The slope of the line connecting the zero point and the reference point will be used to convert from sensor output units to units of PPM. Once the estimated reference point is computed, the slope of the line connecting the zero point and the reference point can be calculated. The SLOPE is computed as shown in Equation 7. The PPM of the calibration gas the sensor was exposed to during the reference calibration is denoted as PPM_CAL. The sysem will obtain PPM_CAL from the working calibration data.

$$SLOPE = \frac{(PPM\_CAL - 0)}{(ADC\_CAL_Z - ADC\_ZERO\_CAL_Z)}$$

Equation 7

The system will check the new conversion values obtained to ensure that they are within the parameters specified by the manufacturer. The SLOPE must be within a valid range specified by manufacturer parameters, and the $ADC\_ZERO\_CAL_Z$ must be below a manufacturer specified threshold. The Sensor Unit will store the new calibration values. The Sensor Unit will inform the Area Monitor that the zero calibration operations have completed.

Specifications for Zero Calibration for the Area Monitor are developed from the requirements. The Area Monitor will initiate the Zero Calibration in response to user action. The Sensor Unit must inform the Area Monitor when the Zero Calibration has completed. The Area Monitor will wait for the Sensor Unit to send the Zero Calibration Complete message, indicating that the Zero Calibration has completed successfully. The Area Monitor will send the zero calibration complete message to the remote computer.

3.1.3 Diagnostic Requirements and Specifications

The system is comprised of three major components, which interact to provide the required functionality of the system. All three of the components must remain in a functional state. If any one of the three components were faulty then the entire system would fail to operate properly. The communications links connecting the components and the proper operation of the Sensor Module are the primary sources of failure in the system.

The system contains two primary communication links connecting the three components. The first, between the Sensor Module and the Sensor Unit, is vital, as the system must read the output of the chemical sensor to calculate a PPM. The remainder of the system requires the PPM to take appropriate actions to provide the required safeguards. The second communication link, between the Sensor Unit and the Area Monitor, must remain intact to allow the system to take required action and to warn personnel in response to elevated PPM readings.

The chemical sensor has a finite lifetime, past which the accuracy of the output decays. The system must alert the user when the chemical sensor has expired and must not allow an expired sensor from being used, as the sensor is no longer useful.

The diagnostic must verify the correct operation of the system and the integrity of the two communication links. The system must inform the user if there is a failure of any diagnostic.

The diagnostic can be initiated from the user at the Area Monitor, the remote computer, or by the system at periodic intervals as shown below in Figure 3-6. The system must initiate a diagnostic periodically to monitor continuously the operation of the system. The user can also initiate the diagnostic, manually, to check the system status at any time via the system user interface or via the remote computer.

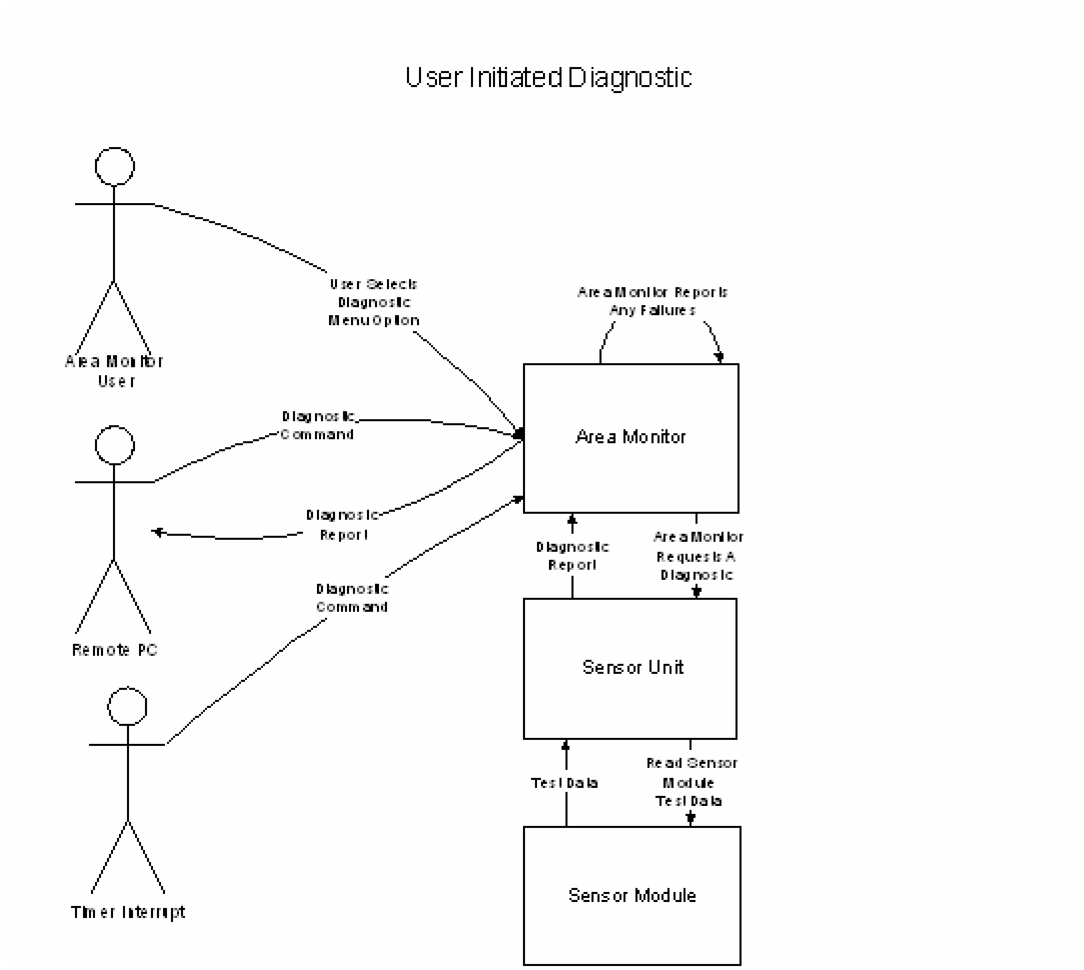

Figure 3-6: Use Case Diagram Showing the User Initiating a Diagnostic or the Periodic Diagnostic.

The Diagnostic must verify that the Sensor Module is not operating past the expiration date, must test the Sensor Module memory integrity and the communications link between the Sensor Module and Sensor Unit, and must test the communications link between the Sensor Unit and the Area Monitor. A report must be generated to indicate the results (pass/fail) of each diagnostic. A diagnostic is successful only if all diagnostic tests pass.

The specifications for the diagnostic for the Sensor Unit are derived from the requirements described above. Figure 3-7 shows the sequence diagram for the periodic system initiated diagnostic. The Sensor Unit must perform the diagnostic tests on the system and generate the pass/fail report.

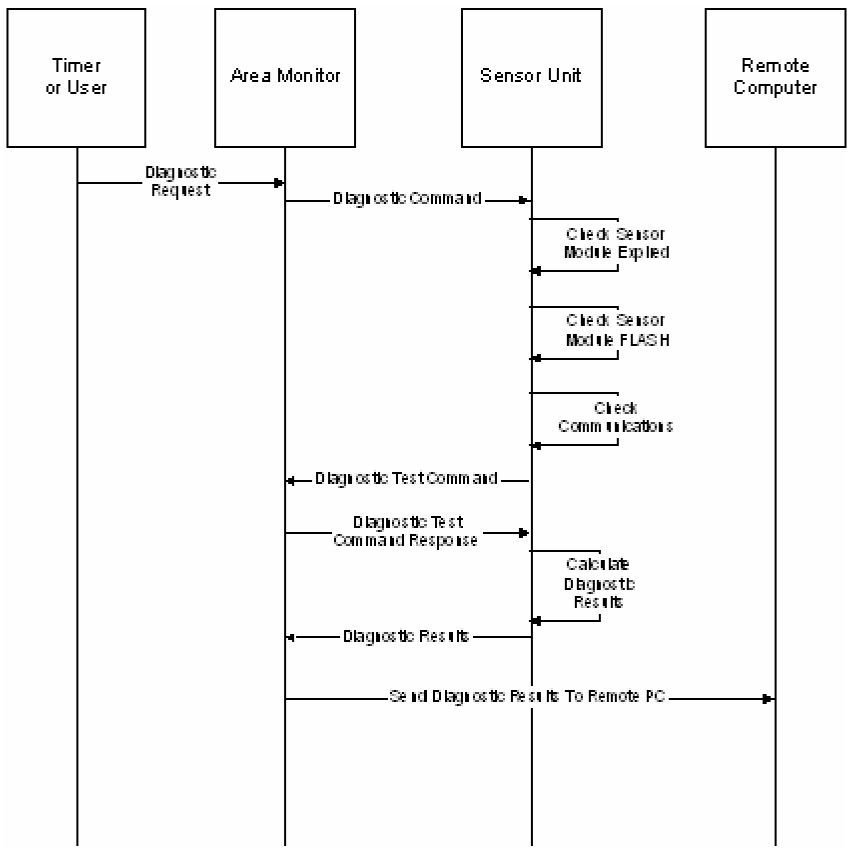

Figure 3-7: Sequence diagram of the diagnostic.

The Sensor Unit must first determine if the Sensor Module has expired. The Sensor Unit will obtain the current month and year, and the manufacture date of the Sensor Module and calculate the number of months from the manufacture date of the Sensor Module. The Sensor Unit will compare the number of months since manufacture to the operational life (in months) of the current Sensor Module to determine if the Sensor Module has expired.

The next diagnostic the Sensor Unit performs must test the integrity of the Sensor Module memory. The Sensor Module will determine the integrity of the Sensor Module memory by reading specific values in the Sensor Module memory and comparing those values to the test values.

Next, the communications link between the Area Monitor and the Sensor Unit must be tested. The Sensor Unit will verify that both the Area Monitor and the Sensor Unit can send and receive messages over the communications link. The Sensor Unit will verify the communications using a special diagnostic handshake protocol. The communication diagnostic is successful if the handshake succeeds. The Sensor Unit will initiate the diagnostic handshake.

After performing the above diagnostics, the Sensor Unit must generate the diagnostic report. The diagnostic report will indicate the success or failure of the individual diagnostics. A diagnostic failure will result in a diagnostic fault to indicate the system is operating incorrectly.

The requirements described previously are used to develop specifications for the diagnostic for the Area Monitor. The Area Monitor must initiate the diagnostic when prompted by the user, or periodically for the system self-diagnostic. The Area Monitor will initiate the periodic diagnostic every 24 hours. The Area Monitor will also initiate the diagnostic when the user selects the diagnostic option from the built-in user interface or the remote computer. The Area Monitor will initiate the diagnostic process by sending the diagnostic command to the Sensor Unit.

The Sensor Unit will perform the diagnostics on the Sensor Module independently, but the Area Monitor will participate in the diagnostic testing the functionality of the communications link between the Sensor Unit and Area Monitor. The Area Monitor will receive the diagnostic test command from the Sensor Unit. The Area Monitor will respond sending the diagnostic test command response to the Sensor Unit, completing the special diagnostic handshake. The Area Monitor will then wait to receive the diagnostic report from the Sensor Unit. The Area Monitor must update the display to reflect the outcome of the diagnostic, and must send the diagnostic report onto the remote computer and remote displays. The Area Monitor will display a fault message to the user if any diagnostic failed.

3.1.4 Configuration Requirements and Specifications

Flexibility of alarm and operational settings allows the system to change operational characteristics to reflect new monitoring regulations, adding versatility to the system. The system must allow the user to update the operational settings. These settings consist of alarm trigger levels and durations for waiting for the observation of a valid calibration point.

The configuration functionality allows the user to change or update the operating settings of the system. The user must be able to update the settings for the alarm limits and the calibration duration periods. The option to change these settings must be available to the user regardless of the system alarm state.

As shown in Figure 3-8, the user selects the configuration option and then enters the new data. The Area Monitor must collect and store the new data, and inform the Sensor Unit that new configuration data is ready. The Area Monitor must send the configuration data to the Sensor Unit. The Sensor Unit receives the new configuration data from the Area Monitor and updates the configuration data. The Sensor Unit must inform the Area Monitor that it has stored the new configuration data. The Area Monitor must inform the remote computer when the Sensor Unit has completed the configuration process.

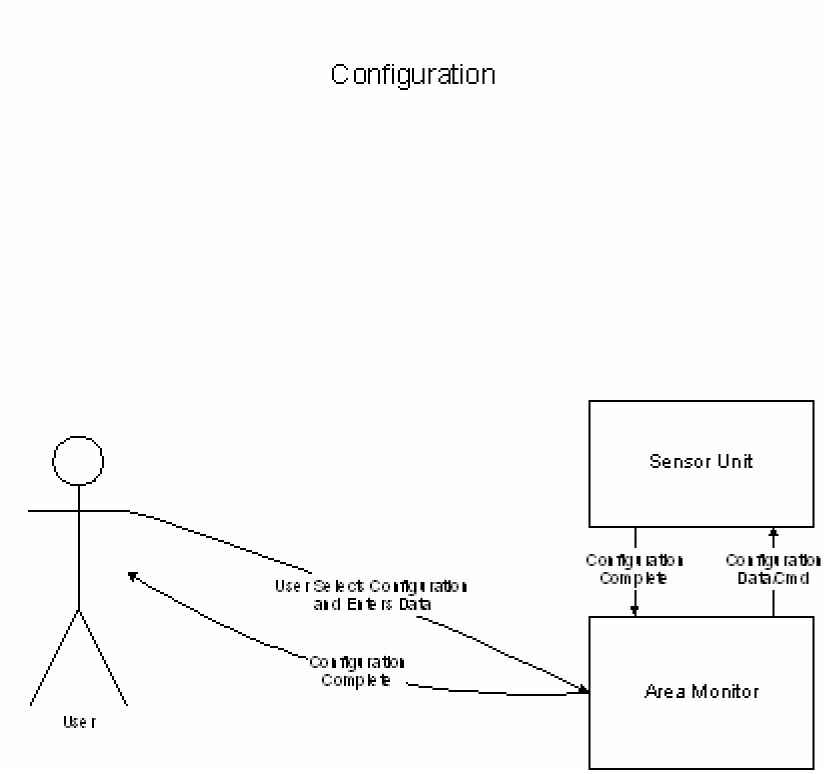

Figure 3-8: Use case diagram showing the configuration operation.

The specifications for the configuration mode for the Sensor Unit follow from the requirements. Figure 3-9 shows the sequence of events for the Configuration routine. The Sensor Unit will perform the Configuration routine in response to the configuration command sent by the Area Monitor. The Sensor Unit must collect and store the new configuration data. The Sensor Unit will collect the data sent from the Area Monitor and will update the configuration settings. After all configuration settings have been updated, the Sensor Unit will send the configuration complete message to the Area Monitor.

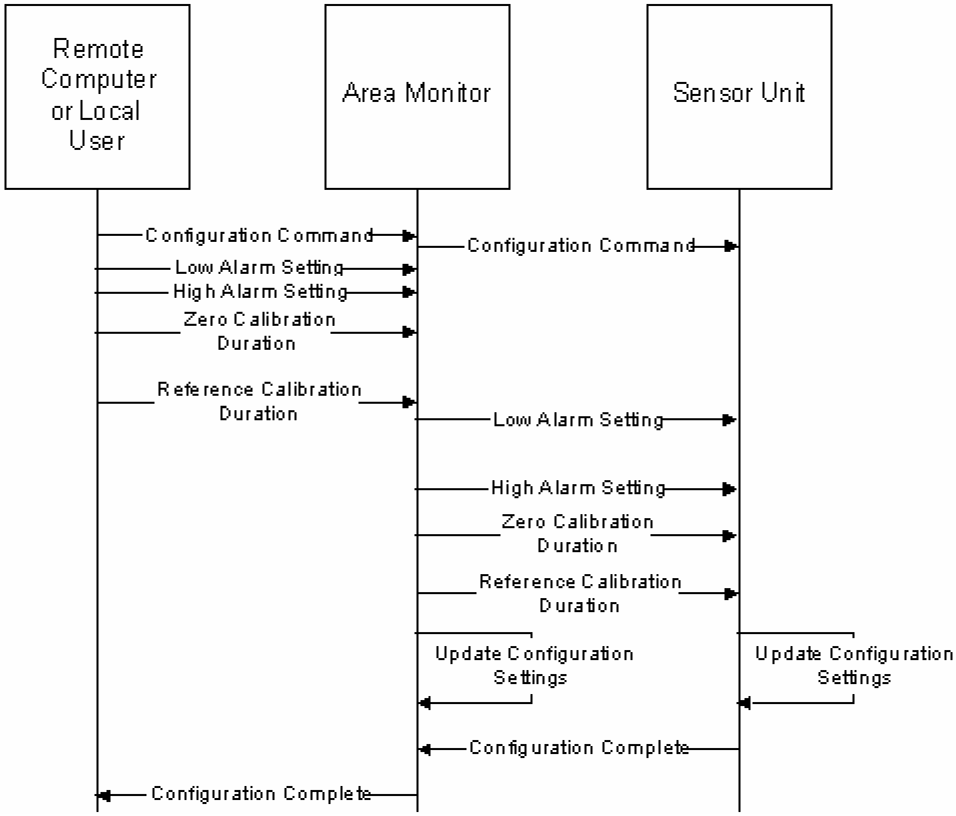

Figure 3-9: Sequence diagram of the Configuration routine.

Specifications are developed for the Area Monitor for the Configuration routine from the requirements. As shown in Figure 3-9, the user must initiate the Configuration routine by sending the configuration command to the Area Monitor. The Area Monitor will then begin to collect the alarm settings and calibration durations data from the user. The Area Monitor must display, to the local user, the current value for each setting and allow the user to change that value. The Area Monitor will maintain the current alarm thresholds and calibration durations in memory. If the user is using the keypad, the Area Monitor must check the entered data against manufacturer specified limits. The low alarm threshold will be below the maximum range of the sensor and the high alarm will be below the maximum sensor range but greater than the low alarm threshold. The zero duration will be between 2 minutes and 5 minutes, and the reference duration will be between 2 minutes and 10 minutes. The Area Monitor will not check the data entered at the remote computer against the specified configuration limits. The Area Monitor will convert any floating-point values into Microchip floating-point format. The Area Monitor will then send the data to the Sensor Unit. The Area Monitor will wait for the Sensor Unit to send the configuration complete message, indicating that the configuration has completed. The Area Monitor will then send the configuration complete message to the remote computer.

3.1.5 Update Limits Requirements and Specifications

Regulations and the site at which the system is installed rarely change after installation. Settings specific to the purpose of the system or to the installation site must be set at installation time and rarely change afterwards. However, providing the ability to update these parameters later makes the system more versatile. The manufacturer also requires a means to set these settings at manufacture time. The system must allow the user to update settings specific to the system installation site and to update the system date and time. Changing these settings allows the system to be updated to reflect changes at the installation site and to regulations. These parameters rarely require change after system installation and are only available for alteration through the remote computer.

The Update Limits functionality provides the capability to adjust the settings specific to the system installation. Figure 3-10 illustrates the Update Limits functionality. The remote computer must send the Update Limits Command followed by the associated data. The Area Monitor must update settings at the Area Monitor, and must forward the Update Limits Command and data to the Sensor Unit. The Sensor Unit must update the settings and must inform the Area Monitor when the Update Limits routine has completed. The Area Monitor after receiving indication that the Sensor Unit has completed the Update Limits routine must inform the remote computer that the Update Limits routine has completed.

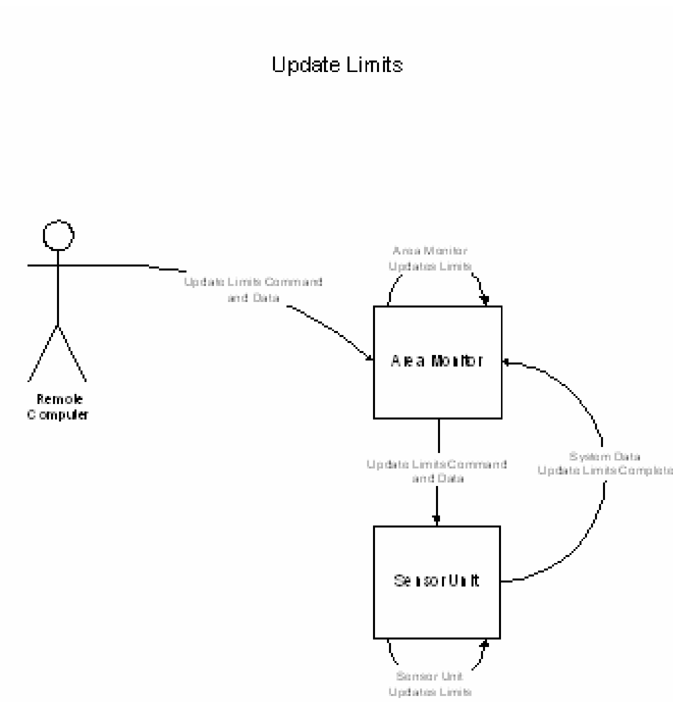

Figure 3-10: Use case diagram for the Update Limits routine.

Specifications for the Update Limits routine are derived from the previously described functional requirements. Update Limits will allow the remote computer user to update settings specific to the system installation. The user can update the altitude of the system used in one of the three PPM compensations. The latching settings specify if the external controls are to be deactivated when entering low alarm from high alarm, or entering no alarm from low alarm. The acknowledge settings indicate if the external control can be disabled when the user presses the alarm silence button. Adjustment of the latching and acknowledge settings which control the operation of the alarm horns and relays with respect the changing alarm status and to the alarm silence button will be provided. The system will store latching and acknowledge settings for the low and high alarm horns and the two external controls. The user will also update the system date and time.

Figure 3-11 shows the sequence of events for the Update Limits routine. The remote computer will initiate the Update Limits routine, sending the update limits command to the Area Monitor. The Area Monitor will forward the update limits command to the Sensor Unit. The Area Monitor will then collect the data sent by the remote computer and will update the altitude, latching, and acknowledge settings. After receiving and storing the new limits data, the Area Monitor will send the limits command followed by the new limits data to the Sensor Unit. The Sensor Unit will receive the new limits data from the Area Monitor. The Sensor Unit will maintain the system date and time. The Sensor Unit will perform the operations necessary to update the system date and time, and will update the altitude, latching and acknowledge values. The Sensor Unit will inform the Area Monitor when the update limits operation has completed. The Area Monitor will inform the remote computer when the Update Limits routine has finished.

3.1.6 Sensor Installation Requirements and Specifications

The Sensor Module is the replaceable portion of the system. The Sensor Module contains the chemical sensor to detect a specific gas, surrounding conditioning hardware, and data specific to the sensor. The system must detect the removal and the insertion of a new chemical sensor. The system must interface with the new sensor to collect the required data. The system must determine when the chemical sensor is ready for operation and must wait to begin normal sampling until the sensor is ready.

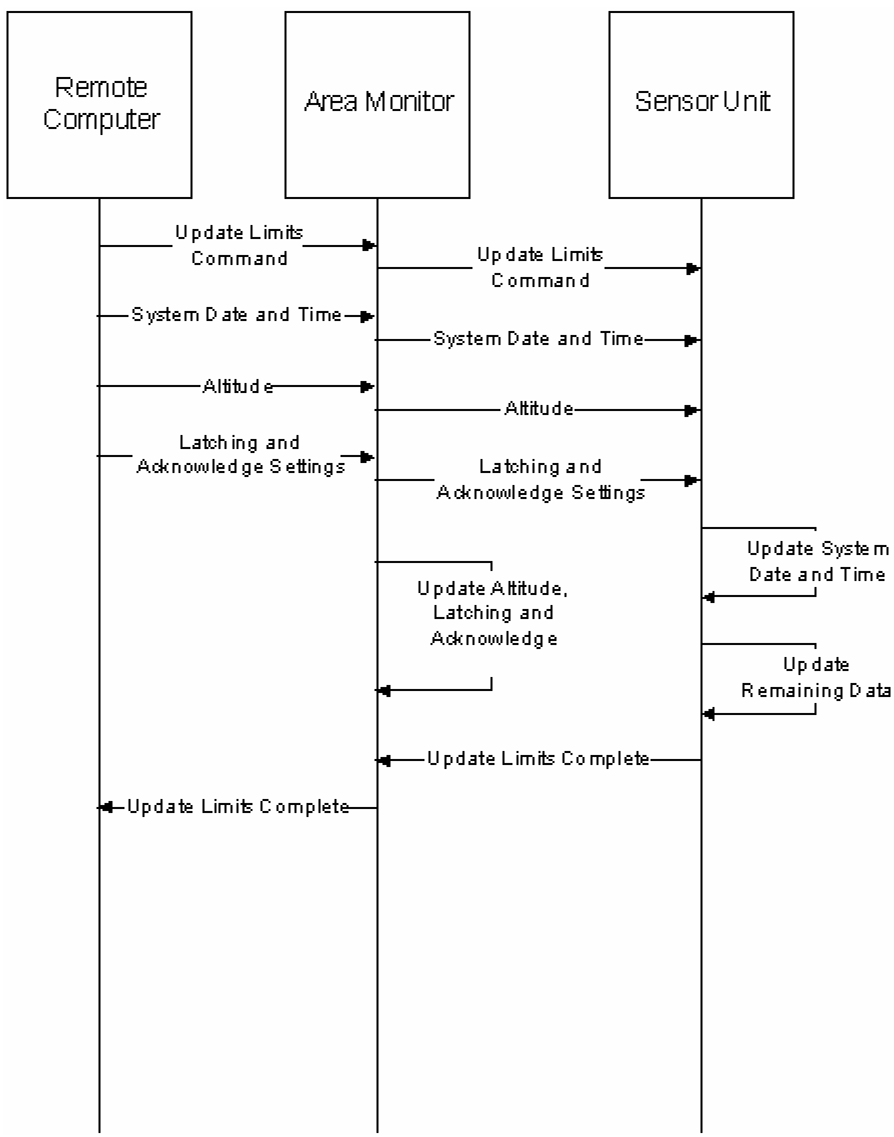

Figure 3-11: Sequence diagram for Update Limits.

The specifications for the insertion of a new sensor for the Sensor Unit and the Area Monitor are derived from the requirements. The Sensor Unit will detect the insertion and removal of the Sensor Module. The Sensor Unit will inform the Area Monitor when the Sensor Module is removed. The Sensor Unit will detect the insertion of a Sensor Module and will inform the Area Monitor that the Sensor Module is present. The Sensor Unit will interface with the newly installed Sensor Module. The Sensor Unit will enter Warm-Up Mode when a Sensor Module is inserted.

The Area Monitor will inform the user when the Sensor Module is missing. The Area Monitor will perform normal operations when the Sensor Module is present. The Area Monitor will inform the remote computer when the Sensor Module is missing, and when the Sensor Module has been detected.

3.1.7 Sensor Warm-Up Requirements and Specifications

A newly installed sensor must adjust to the new environment before providing useful output. The electrochemical reaction controlling the sensor output must stabilize before the system can calculate an accurate PPM. The system must wait for and determine when the electrochemical reaction has stabilized.

Figure 3-12 shows the use case diagram for the Sensor Warm-Up routine. As described in the previous section the user must install the new Sensor Module and the Sensor Unit must detect the new Sensor Module. All required operational data must be downloaded from the Sensor Module. The Warm-Up Mode is responsible for interfacing to a recently installed Sensor Module or when power is restored to the system, and for determining when the Sensor Module is ready for operation. The Sensor Module must be checked to ensure that it has not expired, and the output will be monitored to determine when the Sensor Module is ready for operation.

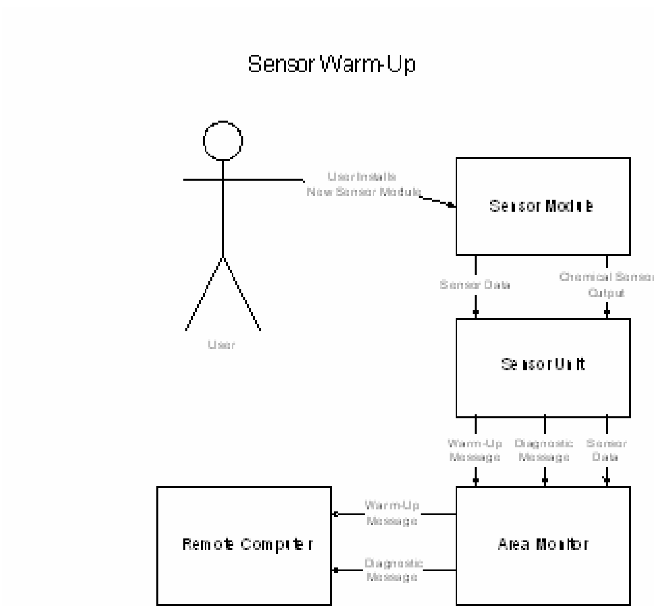

Figure 3-12: Use case diagram of the Sensor Warm-Up functionality.

The specifications for the Sensor Unit for Sensor Warm-Up are derived from the requirements stated previously. Figure 3-13 shows the sequence of events for the Sensor Warm-Up routine. As shown, the Sensor Unit will automatically enter Warm-Up Mode when the new Sensor Module is detected or when power is restored. The Sensor Unit will then inform the Area Monitor that the Sensor Unit is entering Warm-Up Mode. The Sensor Unit must not allow an expired Sensor Module to be used. The Sensor Unit will check if the Sensor Module has already expired. The Sensor Module will obtain the current month and year and the date of manufacture of the Sensor Module from the Sensor Module. The Sensor Unit will use the current date and the Sensor Module manufacture date to determine if the Sensor Module has expired.

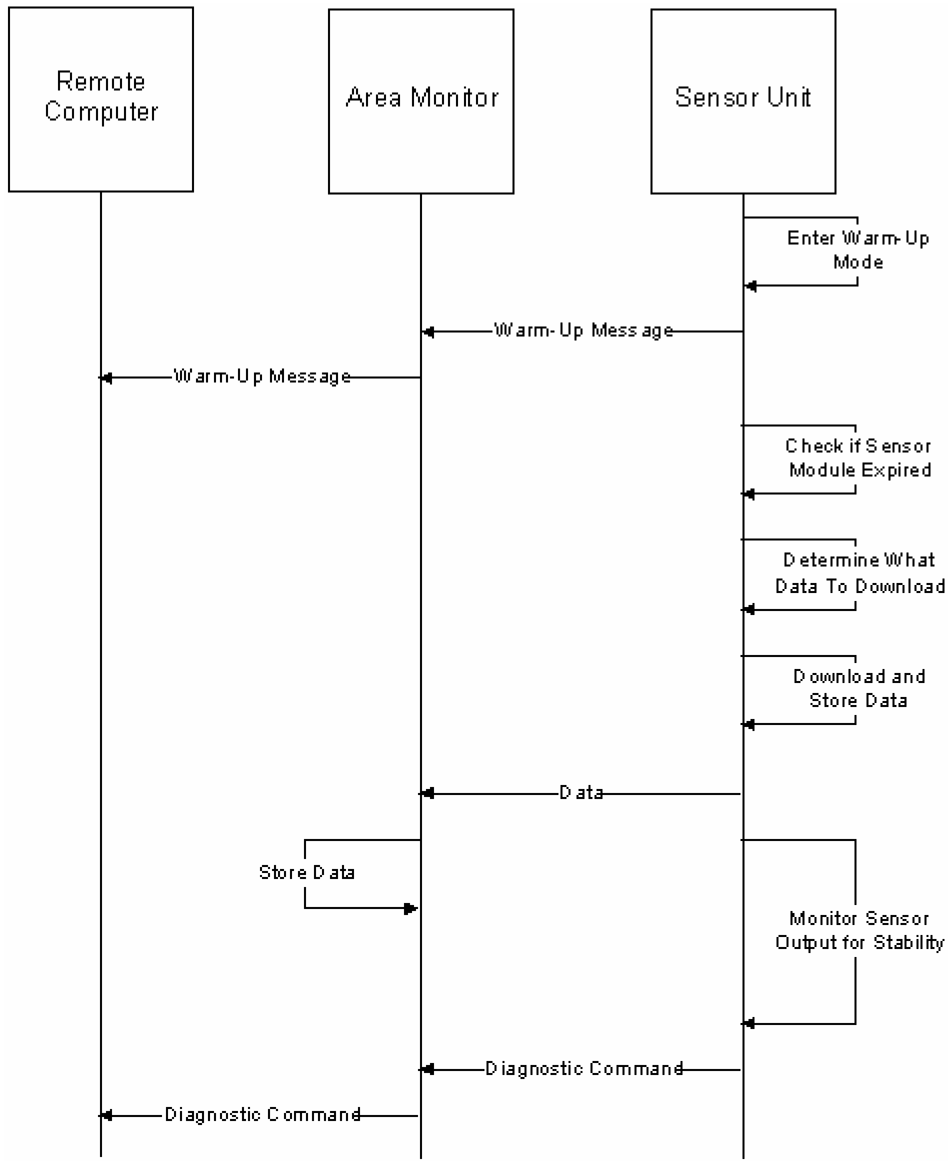

Figure 3-13: Sequence diagram showing the Sensor Warm-Up.

The Sensor Unit will then determine what data must be downloaded based on the serial number and sensor type of the Sensor Module and the serial number and sensor type stored in the Sensor Unit memory. The Sensor Unit will download the Configuration, Calibration, and Sensor Data Tables from the Sensor Module, if the serial number and sensor type are different. The existing data will be used, with the exception of the sensor operational lifetime, which will be recalculated if the serial numbers differ, but the sensor type is the same. No data will be downloaded if both the sensor type and serial number are the same. The Sensor Unit will download and store the required data from the Sensor Module, and will send the required data to the Area Monitor.

The Sensor Unit will then begin to monitor the output of the chemical sensor to determine when the Sensor Module is ready for operation. The Sensor Unit will determine when the Sensor Module is ready for operation by monitoring the output. The Sensor Module will use the magnitude threshold and maximum slope variation parameters specified by the manufacturer to determine when the Sensor Module is ready for operation. The Sensor Module is ready for operation when the output is below the magnitude threshold, and the slope of the first order polynomial least squares fit of ten samples, collected over one second intervals, is no greater than the maximum slope variation. The Sensor Unit must initiate a diagnostic once the sensor output satisfies the above conditions. The Sensor Unit will send the diagnostic command to the Area Monitor to initiate the diagnostic after output satisfies the two tests previously described.

Specifications for the Area Monitor for the Sensor Warm-Up function are derived from the requirements stated above. As shown in Figure 3-13 the Area Monitor will enter Warm-Up mode after receiving the warm-up message from the Sensor Unit. The Area Monitor will send the warm-up message to the remote computer. The Area Monitor will wait for the data to be sent from the Sensor Unit. The Area Monitor will collect all data sent from the Sensor Unit. After all the data has been collected the Area Monitor will store the data in the Area Monitor. The Area Monitor will then wait for the diagnostic command from the Sensor Unit. After receiving the diagnostic command from the Sensor Unit, the Area Monitor will send the diagnostic command to the remote computer. The Area Monitor will then enter diagnostic mode.

3.1.8 System Data Requirements and Specifications

The data required to perform the operations required of the system must be stored in the system. Data required for the calculation of PPM, calibration, configuration, and Sensor Warm-Up, and operation of alarms are needed by several different routines. This data must be available to the system at all times and must be maintained through a power loss.

Maintaining a record of recent system history provides information about the operation of the system over the long-term. This information can be used to identify and correct problems with the system, or to provide a record of the system operation during an investigation. The system manufacturer must be able to view the operational history of the system and requires limited logging of specific information. The system operating parameters and settings must be stored for operation and logging requirements.

The specifications for the Sensor Unit system data are derived from the requirements previously discussed. The Sensor Unit must maintain a record of the current settings and parameters for correct operation of the system functions. The two previous configuration and calibration sets will allow examination of the operation history of the system. The Sensor Unit must maintain the previous two configuration and calibration data sets, in addition to the working data sets through a power loss.

The Sensor Unit will contain four tables containing working data, and four tables containing system history data. The four working tables will contain the data required for the calculation of PPM, configuration, calibration, alarm operation, and Sensor Warm-Up. Four tables will be used to store the information, the configuration table, the calibration table, the limits table, and the sensor data table.

The configuration table will hold the alarm and calibration duration settings. The alarm settings are the threshold values used to determine the alarm state of the system after calculating a PPM. The system will contain two alarm thresholds, the low alarm threshold and the high alarm threshold. The low alarm threshold denotes the boundary between the no alarm status and low-alarm status, and the high alarm threshold denotes the boundary between the low alarm status and the high alarm status. The calibration duration settings specify the amount of time the system must provide for a calibration point to be observed. The system will maintain two calibration durations, the zero duration for the Zero Calibration and the reference duration for the Span Calibration. The zero duration and reference duration will specify the time limit in minutes for a valid calibration to be observed for the Zero Calibration and Span Calibration respectively. The user changes the configuration table during a configuration.

The calibration table will contain all information entered and calculated during a calibration. The system requires the values calculated during a calibration to calculate the PPM and the manufacturer requires the user entered values for the span calibration and date of calibration as part of the historical data. The two calibration points found in each of the calibrations are needed for the Zero Calibration and for the calculation of the PPM. The calibration table will contain two sets of calibration points. The first set is the points found during a Span and Zero Calibration, and the second set is the points found from a Zero

Calibration. The temperature of the most recent calibration must be stored as it is used to compensate the PPM value and for the estimation of the reference point during a Zero Calibration. The system will store the temperature of the most recent calibration temperature. The slope of the line connecting the two calibration points must be stored so that the sensor output can be converted into units of PPM. The system will store the slope of the line connecting the two calibration points. The user must provide the scale factor and the reference PPM for the reference calibration. The system will store the reference PPM, the scale factor, and the date of calibration for logging purposes. The calibration table is updated when the user performs a calibration.

The limits table will contain information based on the location and purpose of the system installation. The altitude must be known to perform the altitude compensation during the calculation of the PPM. The altitude will be stored in the limits table. The latching and acknowledge settings control the operation of the alarms and external controls. The latching and acknowledge settings will be stored in the limits table. The user must be able to update the system clock. The limits table will maintain a copy of the most recent date and time the system clock was set to. The limits table can be changed only be the remote computer.

The sensor data table will contain additional data that is specific to the sensor. The sensor data table must contain information specific to the Sensor Module. The system will store this information in the sensor data table. The table will contain information needed by the remote computer, the sensor operational and shelf life, the manufacture date of the Sensor Module, and the sensor serial number and type. The table will also contain the parameters describing the piecewise linear fits to the three compensation curves used during the computation of PPM. Parameters for finding the calibration points and for evaluating the warm-up tests will be stored in the table. The data in the sensor data table is specific to the Sensor Module and only replacement of the Sensor Module will change this data.

The Area Monitor must check all input entered via the built-in interface for validity. The Area Monitor will verify all information entered during the configuration, and will validate the reference PPM entered during the Span and Zero Calibration. The Area Monitor must maintain a current record of the system operational settings for the purposes of the man-machine interface. The Area Monitor will also contain the four working tables used by the Sensor Unit. The Area Monitor will receive these tables during the Sensor Warm-Up from the Sensor Unit, and will update the tables when during the Configuration, the Zero Calibration, the Span and Zero Calibration or the Update Limits routines.

3.1.9 Download System Data Functional Requirements

The remote computer must record and compute the long-term exposure statistics, and provide the ability to perform maintenance operations. To provide these capabilities the remote computer must have the same data as the system. The remote computer must be able to request the system data from the Area Monitor. Specifically this data is needed when the remote computer is first connected to the system, when the remote computer must resynchronize with the system, or to examine the system historical data.

The Download System Data functionality provides the capability for the remote computer to obtain a copy of all the system data stored in the non-volatile memory. All the system operation data maintained by the system will be sent to the remote computer.

As shown below in Figure 3-14, the remote computer must send the download system data request to the Area Monitor. The Area Monitor must then request the system data from the

Sensor Unit. The Sensor Unit must send all system operational and historical data to the Area Monitor. After receiving all system data from the Sensor Unit, the Area Monitor must transfer the system data to the remote computer.

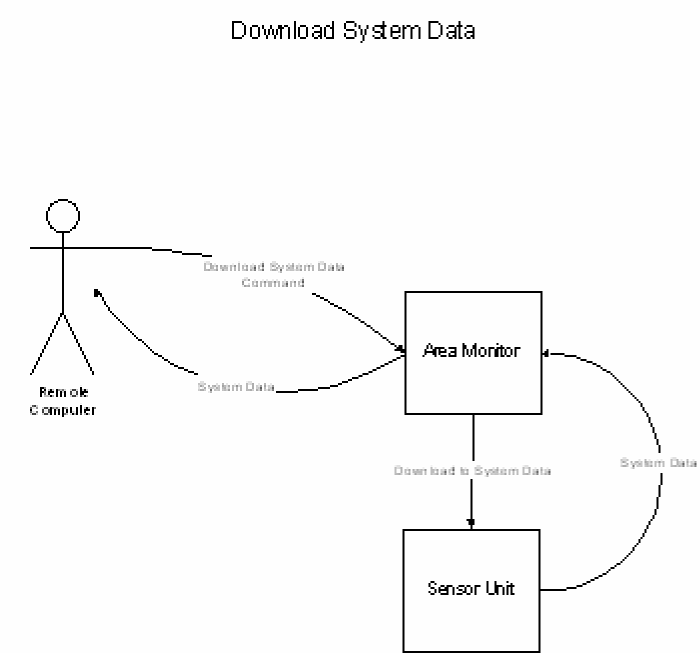

Figure 3-14: Use Case of download of system data routine.

Specifications for the Download System Data functionality are developed for the Sensor Unit and the Area Monitor based on the requirements above. The Sensor Unit contains the current working and operational history system data. The Sensor Unit will send this data to the Area Monitor in response to the download system data command. All floating-point values will be stored in Microchip floating point format for use within the Sensor Unit and must be converted into IEEE-754 floating-point format. The Sensor Unit will convert all floating-point data to IEEE 754 floating-point format and will send this data to the Area Monitor.

The remote computer, as shown in Figure 3-15, must initiate the Download System Data routine sending the download system data command to the Area Monitor. After receiving the download system data command from the remote computer, the Area Monitor will send the download system data command to the Sensor Unit. The Sensor Unit will convert all floating-point data to IEEE 754 format and send the system data to the Area Monitor. The Area Monitor will collect the system data, sent from the Sensor Unit. After all data has been received from the Sensor Unit, the Area Monitor will send the system data to the remote computer.

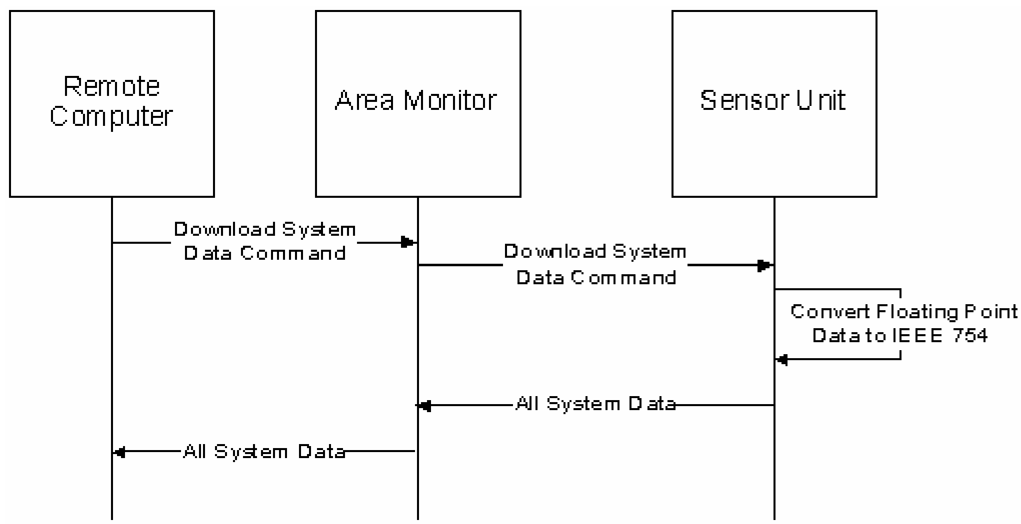

Figure 3-15: Sequence diagram showing the Download System Data routine.

3.1.10 Request System State Functional Requirements

The remote computer may occasionally lose synchronization with the operation of the system. When this occurs, the remote computer may not be able to interpret properly the data received from the Area Monitor and the remote computer could send commands and/or data that could result in a system crash. Therefore, the system must provide a means for the remote computer to resynchronize with the system.

The system must provide a means to allow the remote computer to request and subsequently receive the current system operating state. In response to this request, the system must provide the remote computer with the current operating state of the system in response to this query.

The remote computer, as shown in Figure 3-16, must request the system state from the Area Monitor. The Area Monitor must then send the current state information to the remote computer. The remote computer will use this feature to synchronize operation with the operation of the system.

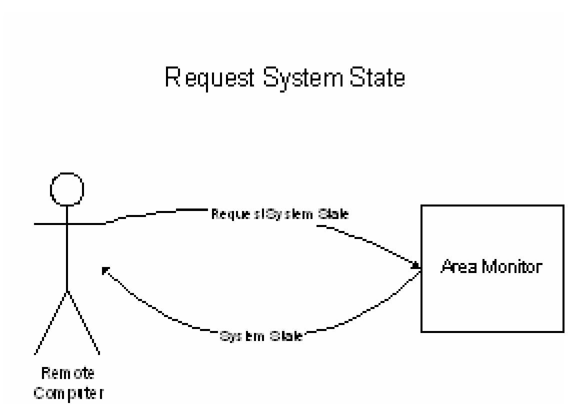

Figure 3-16: Use case diagram showing the request system state operation.

Specifications for providing the remote computer the means to resynchronize with the system are derived from the requirements listed above. Figure 3-17 shows the sequence diagram for the request system state functionality. The remote computer will initiate the operation by sending the request system state message to the Area Monitor. The Area Monitor will then respond by sending the state information to allow the remote computer to synchronize with the system.

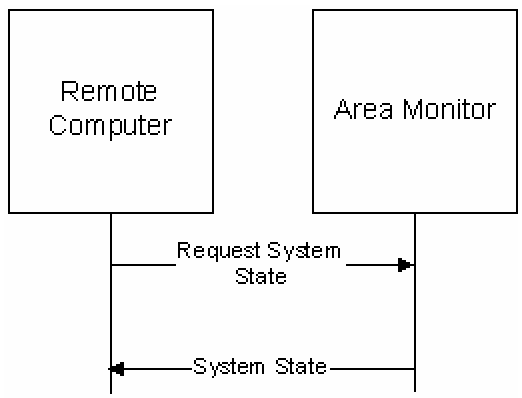

Figure 3-17: Sequence Diagram for Request System State functionality.

3.1.11 Man-Machine Interface

The system must provide a built in man-machine interface and a protocol to control operation from a remote computer. The man-machine interface must provide the ability to perform all required system maintenance operations. The remote computer must be able to control operation of the system. The system must provide a communication protocol to communicate with the remote computer.

The built in system Man-Machine interface must consist of a visual display for display purposes, and a keypad for user input. The Man-Machine Interface must display information about the PPM, current system status and available options to the user. The user must also be able to perform the operations necessary for operation of the system using the built in Man-Machine interface. The user must be able to enter or update system data using the built-in man-machine interface. The display must be readable from a distance.

The remote computer will allow a user to execute, remotely, system functions and to monitor the status of the system. The remote computer and the Area Monitor will interact using the system communication protocol. The Area Monitor must receive, interpret, and act on the communication from the remote computer. The remote computer will provide long-term logging of system operation.

The built-in man machine interface provides interfaces for performing the required system maintenance operations. The interface must provide the user the ability to update current system configuration settings, to initiate a zero calibration, to perform a span and zero calibration, to initiate a diagnostic, and to initiate action required to clear an outstanding fault. The remote computer will provide all of the above functionality, and provides the user with the ability to change settings specific to the system installation. The Area Monitor must provide a means to allow the remote computer to request the current system state for synchronization purposes, and the ability to download all data stored by the system.

Specifications for the built-in interface and the remote computer communications protocol are developed from the requirements above. The built-in man-machine interface will use a sequence of menus to provide the required interaction. The Area Monitor and the remote computer will use the system protocol to communicate. The Area Monitor will respond to requests from the remote computer and will send status information to the remote computer.

The current operating state and the current alarm status will determine what options are available to the user. The Area Monitor will allow only one operation to occur at a time. For example, if a configuration were initiated, a zero calibration could not be initiated until the configuration finished. Hence, once an operation has started the user must finish that operation before starting another operation. Therefore, the Area Monitor will start operations only when the system is in Normal Operation Mode, Alarm Mode, or Fault Mode. The built-in interface will also provide the user the means to enter data for a Configuration or a Span and Zero Calibration. The Area Monitor will display the current data setting and provide the ability for the user to change that value.

Normal Operation Mode is the normal operating state of the system, as the alarm status is no alarm and no outstanding faults. The Area Monitor will allow the user to start any operation in Normal Operation Mode.

Alarm Mode describes the system when there are no outstanding faults and the alarm status is low, high, or over-range. The Area Monitor will only allow the user to perform a configuration or a request of system state in Alarm Mode.

Fault Mode indicates that a fault has occurred and is currently outstanding. The Area Monitor will only allow the user to request the system state, or to perform the operation required to clear the outstanding fault.

The built-in man-machine interface will use a menu system to display the available options to the user and a keypad to accept user input. The display will consist of a 4-character display, capable of displaying information in green, yellow, and red. The keypad will consist of

5-buttons up, down, left, right, and enter. The display will consist of four characters. The user can initiate any operation when the system is in Normal Operation Mode. Figure 3-18 shows the menu transition diagram when the system is in Normal Operation Mode. Menu mode will allow the user to select an operation to perform, while normal display will display the current PPM. The Area Monitor will display the current PPM reading unless the user enters menu mode. Menu mode presents the user with the available options.

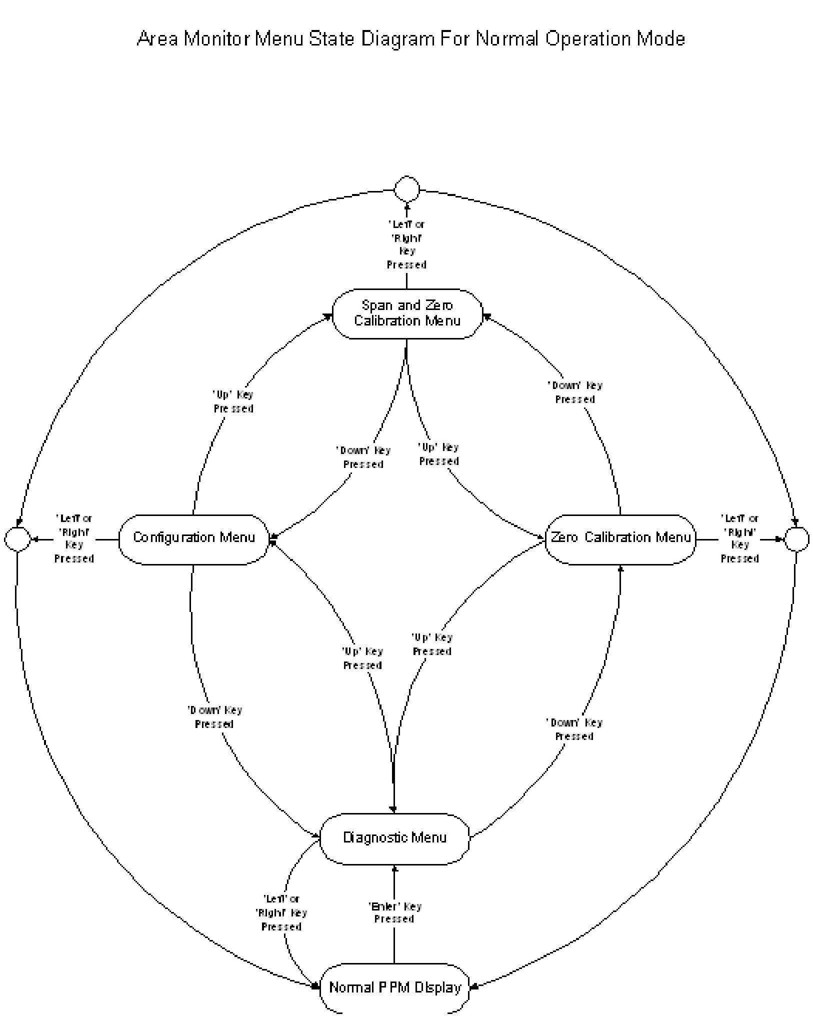

Figure 3-18: Normal Operation Mode menu state diagram.

When the system is in Normal Operation Mode the user can initiate a configuration, a zero calibration, a span and zero calibration, and a diagnostic. The user will enter menu mode by pressing the enter key. Once in menu mode pressing the up and down keys will cycle through the menu options in opposite directions. Pressing the left or right keys will return to normal display mode from any menu option. Pressing the enter key will initiate the currently selected option. The Area Monitor will return to the normal display menu after completion of each operation.

When in Alarm Mode, the Area Monitor will only allow a Configuration to be initiated from the built-in user interface. Figure 3-19 shows the state transition diagram for the Alarm Mode menu system. The Area Monitor will normally display the PPM when in Alarm Mode. The user will press the enter key to enter menu mode. Since configuration is the only operation that can be performed, while in Alarm Mode the up and down keys will not cause any transitions in the Alarm Mode menu. Pressing the enter key when in menu mode will initiate a configuration. Pressing the left or right keys when in menu mode will return to the alarm PPM display.

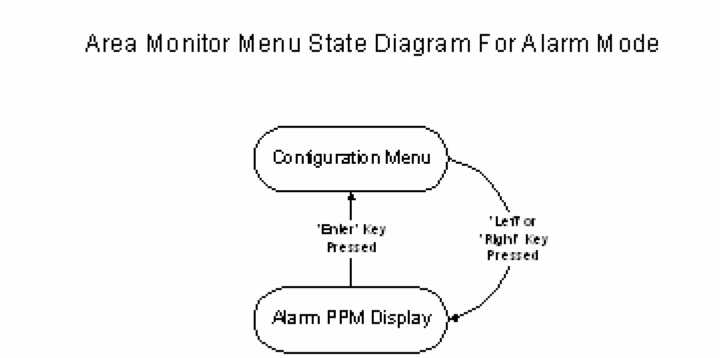

Figure 3-19: Alarm Mode menu state diagram.

When the system is in Fault Mode only the operation required to clear the outstanding fault can be initiated. Pressing the enter key will initiate the required operation, and the other keys will have no effect. The display will be red and flashing when in Fault Mode.

The remote computer and Area Monitor will communicate using the system protocol. The remote computer will send the command to initiate the operation to the Area Monitor. The Area Monitor will initiate the operation if possible. While the remote computer is performing an operation the Area Monitor will display the current operation being performed to the user and will prevent another operation from being initiated until the current operation finishes. The Area Monitor will send status information and PPM readings to the remote computer.

3.1.12 System Faults Requirements and Specifications

The ability of the system to function properly can be effected by a number of events. The system must provide a means to detect these events, or faults, capable of preventing normal operation. The system must identify and inform the user of any detected faults. The system must provide the user the ability to take action to correct the faults.

The faults described in this paragraph can adversely affect system functionality and must be monitored for. The Sensor Module must be present for the system to function. The system must inform the user if the Sensor Module is not present. The system must alert the user to a failure of any diagnostic. The system must alert the user to the failure to find a calibration point within the allotted time, or if the newly calculated calibration values are out of range. The system must inform the user when the Sensor Module has expired and when the integrity of the non-volatile memory may have been compromised.

The requirements above allowed the development of the specifications for Fault Mode. Fault Mode will inform the user of a currently outstanding fault. The system will provide the user with the currently outstanding fault and will allow the user to perform the action required to clear the outstanding fault. The Area Monitor will display the outstanding fault.

The system will monitor for and inform the user of the following faults.

- Sensor Module Missing – Sensor Module is not detected or not connected properly. A Sensor Module must be inserted to clear this fault.
- Diagnostic Failed – One or more diagnostics have failed. A diagnostic that passes must be performed to clear this fault.
- Zero Calibration Failure – The zero point was not found in the allotted time, the slope is outside the manufacturer limits, or the zero point is above the manufacturer's maximum zero point. A successful Zero Calibration must be performed to clear this fault.
- Span Calibration Failure – The zero or reference point where not found in the allotted time or the slope was outside the manufacturer specifications. A successful Span and Zero Calibration must be performed to clear this fault.
- Sensor Module Past Specified Lifetime – The newly inserted Sensor Module is operating past the specified lifetime. A Sensor Module that has not expired must be installed to clear this fault.
- Loss of Power – Power was lost during an update to the non-volatile memory. The user must acknowledge this fault to clear it. The user should perform the operation that was being performed when power was lost to ensure that valid data is stored.

3.2 TEST PLAN

This section describes the tests required to verify that the system meets the requirements and specifications presented in the preceding sections.

- The software correctly transitions between Alarm Mode and Normal Operation Mode based on the PPM value and alarm thresholds.
- Both a passing and a failing diagnostic can be performed.
- A configuration can be performed and the working configuration table is updated to reflect the new configuration data and that the previous configuration tables maintain the required configuration history.
- A zero calibration produces a 0-PPM reading around the new zero point and updates to the working and previous calibration tables are performed.
- A span and zero calibration can be performed, giving correct PPM readings after the calibration around the two calibration points. The calibration tables must be checked to ensure that the working and previous calibration tables are properly updated.
- The system executes the Sensor Warm-Up correctly when power is supplied.
- Faults are properly displayed and cleared.
- The remote computer can initiate operations.
- The Download System Data routine provides the correct data to the remote computer.
- The Request System State provides the correct state information to the remote computer.
- The Update Limits routine correctly modifies the existing data in the limits table.

4.0 SOFTWARE DESIGN

4.1 DESIGN OF THE TOP-LEVEL SOFTWARE

The requirements and specifications developed in the previous section provide a basis for developing the software to control the Area Monitor and the Sensor Unit. The Sensor Unit will perform the majority of the calculations in the system, and the Area Monitor will control the man-machine interfaces. The Area Monitor will control the interaction between the user and the Sensor Unit and the remote computer.

Communication or user input can happen at any time, the software must be able to handle input essentially instantaneously. Two methods to handle input are polling and interrupt service routines (ISR). In a polling scheme, each input device is checked for received input periodically, an ISR on the other hand is executed when the input is detected. Polling does not provide for the immediate response to an input that the ISR provides. For this reason, ISRs will handle incoming RS-232 communications and user input from the built-in man-machine interface. The use of ISRs ensures the quickest possible reaction to input.

In addition, to the interrupts handling the RS-232 communications, a timer interrupt, a keypad interrupt, and an alarm silence button interrupt will be used. The time keeping interrupts trigger the PPM calculations and samples during the warm-up tests and during calibrations. The keypad interrupt handles the keypad input, informing the main software what key was pressed, and updates the data to be displayed, during data entry. The alarm silence button will disable the alarms and relays based on the current system settings and alarm state when the button is pressed.

Because the system is a distributed system, each part must remain synchronized with the actions of the other. Losing synchronization will cause the system to fail because messages and data sent between the Sensor Unit and Area Monitor may not be interpreted properly. The software on each node will maintain the synchronization with the other node by sending acknowledge messages during critical transitions.

The communication links between the Sensor Unit and Area Monitor and between the Area Monitor and remote computer use RS-232 without the handshaking part of the standard. The ISR will use the current system state to determine how to interpret the received message and what action to take.

The processors are operating much faster than the RS-232 communication links. Therefore, the processor sending the message may return to the execution of code before the receiving processor processes received the message. The sending processor can switch state and send more messages based on the new state. The lack of the RS-232 handshaking signals introduces this possibility. If this occurs the unit receiving the messages may not have sufficient time to change the state, and the ISR will interpret the received message incorrectly. To prevent this from occurring, synchronization points will be placed in the software after critical transitions. The part that could possibly lag behind the other will send an acknowledge message after the transition is completed.

The software on each portion of the distributed system will be modeled as a finite state machine. The transition between states provides the means to synchronize the operation of both portions. The top-level operating states and transitions are defined based on the requirements and specifications of the system functionality discussed in the preceding section. Each operation will have its own top-level state. Each of these top-level states may have internal state machines to complete the required task. Those operations having multiple states will have internal state machines within the top-level state. Figure 4-1 shows the top-level state machine for the whole system. The Sensor Unit and the Area Monitor both have state machines implementing the system state machine.

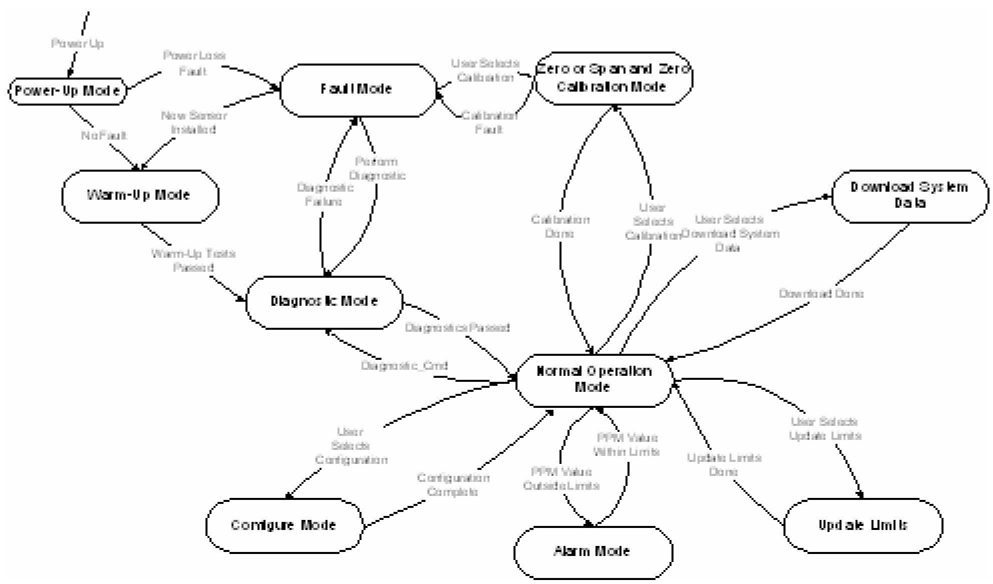

Figure 4-1: Top-level state system machine.

As previously described, the system will compute the PPM once per second and will allow the user to initiate any operation while the alarm state is no-alarm. The Normal Operation Mode combines the above functionality into one state, and is the normal operating state for the system. In this state, there are no outstanding faults and no alarm conditions. The system will regularly update the PPM as described in Section 3.1.1 and the user may initiate any operation from this state, as described in Section 3.1.11.

When the alarm-state is other than no-alarm, the user can only initiate a configuration, but the system will continue to update the PPM every second and operate the alarms. A different state is needed when the alarm status is no-alarm because of the limitation on the operations that can be performed and the need to activate the alarms. Alarm Mode indicates that the system has detected and reacted to an alarm condition. The Alarm Mode state updates the PPM, operates the alarms, and places the user interface in a state allowing only configuration. There are three different available alarm statuses in alarm mode; low alarm, high alarm, and over-range. Over-range and high alarm act similar in that both the relays and alarm horns are active. In low alarm state only the low alarm horn and low alarm relay are active. The low alarm state is defined as a PPM value below the high alarm limit but above the low alarm limit. High alarm is defined as a PPM value above the high alarm limit but below the range of the system, and the over-range alarm represents a PPM value greater than the range of the system. The system will regularly update the PPM as described in Section 3.1.1. The user may initiate only a configuration when the system is in alarm mode.

Zero Calibration Mode will allow the user to perform a Zero Calibration described in Section 3.1.2.2. The user may need to connect equipment to the system to perform a Zero Calibration. The time taken to connect the calibration equipment to the system must not deduct from the zero duration time limit. The Area Monitor will wait for the user to instruct the unit to begin monitoring before initiating the monitoring for the zero point.

Span and Zero Calibration Mode will allow the user to perform the Span and Zero Calibration operation described in Section 3.1.2.1. As previously discussed, the user will be prompted to inform the system when to start monitoring the sensor output for each calibration point.

Configuration Mode will allow the user to perform the Configuration function described in Section 3.1.4. Update Limits Mode will allow the user, via the remote computer, to perform the Update Limits operation described in Section 3.1.5.

Power-Up Mode will inform the user that power has just been restored to the system. The actions of the system in Power-Up mode are determined entirely by the status of the system at power up. The check for the Loss of Power Fault will be performed in this mode. This is the only entry point for the Loss of Power Fault. Warm-Up Mode will perform the Sensor Warm-Up function described in Section 3.1.7. Fault Mode will provide the user with the currently outstanding fault and allow the user to perform the action required to clear the outstanding fault as described in Section 3.1.12.

Download System Data Mode will perform the Download System Data function described in Section 3.1.9 to provide the remote computer with the system data. Synchronize Mode will perform the request system state functionality described in Section 3.1.10 to allow the remote computer to synchronize with the system.

4.1.1 Sensor Unit Top-Level Software

The Sensor Unit software must implement the system state machine previously described. The state machine illustrated in Figure 4-2 shows the top-level software state diagram of the Sensor Unit software implementing the system state machine for the Sensor Unit functions. The

Sensor Unit software will consist of ten top-level states. The Sensor Unit will transition between states based on received commands and data from the Area Monitor and events internal to the Sensor Unit.

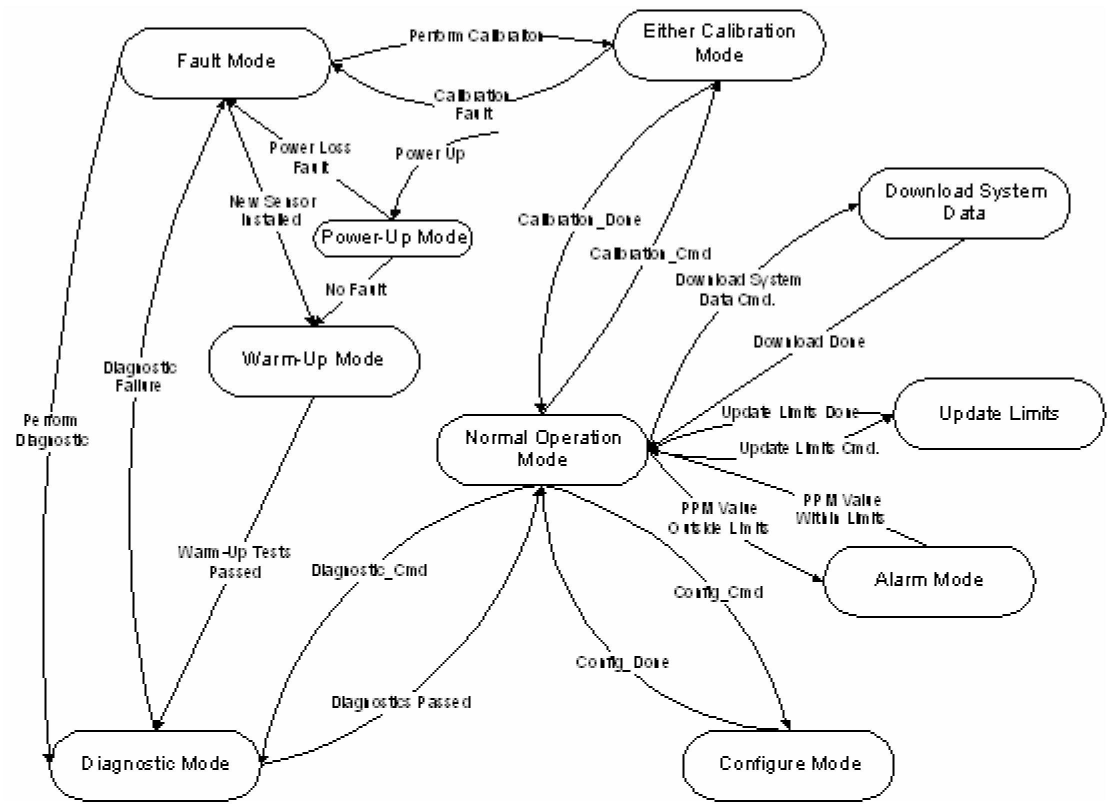

Figure 4-2: Top-level Sensor Unit state diagram.

The Sensor Unit will normally operate in the Normal Operation Mode. The Sensor Unit will calculate the PPM when requested by the Area Monitor and will monitor for the occurrence of a fault. After calculating the PPM, the Sensor Unit will determine the alarm status of the system using the new PPM as described in Section 3.1.1. If the alarm status is found to be no-alarm the Sensor Unit will remain in Normal Operation Mode, if the alarm status is found to be low, high, or over-range the Sensor Unit will enter Alarm Mode. The Sensor Unit will enter one of the other top-level states in response to a command received from the Area Monitor. The top-level state entered is the state to perform the requested operation. The Sensor Unit RS-232 ISR will cause a state transition when a command is received from the Area Monitor.

The Sensor Unit will enter Alarm Mode when the alarm status is low, high, or over-range status. The Sensor Unit will evaluate the alarm status after calculating the new PPM. The Sensor Unit will return to Normal Operation Mode if the alarm status is no-alarm, otherwise the Sensor Unit will remain in Alarm Mode. The Area Monitor RS-232 ISR will cause the transition into Configuration Mode if the configuration command is received from the Area Monitor.

The Sensor Unit will enter Configuration Mode from Normal Operation or Alarm Mode, after the Area Monitor RS-232 ISR receives the configuration command. After entering Configuration Mode the Sensor Unit will send the acknowledge message to the Area Monitor indicating the Sensor Unit has transitioned into Configuration Mode, and preventing the Area Monitor from sending the configuration data before the Sensor Unit is ready to receive it. The Sensor Unit will wait for the Area Monitor to acknowledge the configuration complete ok message at the end of the configuration. This prevents the Sensor Unit from entering Normal Operation Mode before the Area Monitor.

The Sensor Unit remains in Update Limits Mode and Download System Data Mode until the executing routine has completed. The Sensor Unit will enter Update Limits Mode after receiving the update limits command, and enters Download System Data Mode after receiving the download system data command. Once the currently executing routine completes the Sensor Unit will return to Normal Operation Mode. For these two modes, only one portion will send data. The portion sending data will space the sending of data internally to prevent a data byte from being overwritten on the receiving end. Therefore, synchronization in these two modes is not needed.

The Sensor Unit will enter Zero Calibration Mode in response to receiving the zero calibration command. Upon entering Zero Calibration Mode the Sensor Unit will send the acknowledge message to the Area Monitor indicating that the Sensor Unit has entered Zero Calibration Mode from Normal Operation. This prevents the Area Monitor from sending the zero calibration start command before the Sensor Unit is ready to perform the zero calibration. The Sensor Unit will wait for the Area Monitor to acknowledge the zero calibration complete command, sent after completing the zero calibration, before entering Normal Operation Mode. The Sensor Unit will enter Fault Mode if a fault is detected during the zero calibration.

The Sensor Unit will enter Span and Zero Calibration Mode from Normal Operation Mode after receiving the span and zero calibration command. The Sensor Unit will send the acknowledge message to the Area Monitor indicating that the Sensor Unit has transitioned into Span and Zero Calibration Mode. This prevents the Area Monitor from sending commands the Sensor Unit is not ready to handle. The Sensor Unit will wait for the Area Monitor to acknowledge receipt of the zero calibration complete and span calibration complete messages before transitioning to the next state, preventing the Sensor Unit from moving too far ahead of the Area Monitor. The Sensor Unit will return to Normal Operation Mode if the calibration is successful, or will enter Fault Mode if a fault is detected during the calibration.

The Sensor Unit will enter Diagnostic Mode after completing Warm-Up Mode, or in response to the diagnostic start command sent from the Area Monitor. The Sensor Unit will wait for the Area Monitor to acknowledge before performing the sensor diagnostics. The Sensor Unit will also wait for the Area Monitor to acknowledge the send diagnostic report command to prevent the diagnostic report from being sent to quickly. The Sensor Unit will wait for the Area Monitor to acknowledge the diagnostic report, preventing the Sensor Unit from entering Normal Operation Mode before the Area Monitor. The Sensor Unit will return to Normal Operation Mode after successful completion of all diagnostics. The Sensor Unit will enter Fault Mode if one or more diagnostics fails, or is fault is detected.

The Sensor Unit will enter Power-Up Mode when power is first supplied to the system. The Sensor Unit will enter Fault Mode if the Sensor Module is not present, or if a Loss of Power Fault is detected, otherwise the Sensor Unit will enter Warm-Up Mode.

The Sensor Unit will enter Warm-Up Mode from Power-Up Mode or from Fault Mode. The Sensor Unit will remain in Warm-Up Mode until the Sensor Module passes the warm-up tests, or a fault is detected. The Sensor Unit will wait for the Area Monitor to acknowledge the entry into Warm-Up Mode before performing any functions. This is to prevent the Sensor Unit from sending the data to the Area Monitor before the Area Monitor is ready to accept the data. The Sensor Unit will enter Diagnostic Mode after completing the warm-up tests or will enter Fault Mode if a fault is detected.

The Sensor Unit will enter Fault Mode from any state if the Sensor Module is detected as missing or in response to the failure of an operation. Each fault requires the user to take different action to clear. The state the Sensor Unit enters when the user performs the required action depends on the current outstanding fault. The Sensor Unit will enter Power-Up Mode from a Sensor Module Missing, a Sensor Module Past Lifetime Fault, or when a new Sensor Module is inserted. If a Loss of Power Fault is detected during Power-Up Mode, the Sensor Unit will enter Warm-Up Mode after the user acknowledges the Loss of Power Fault. The Sensor Unit will acknowledge the receipt of the acknowledge loss of power fault, and will then wait for the Area Monitor to acknowledge before entering Warm-Up Mode. This synchronizes the entry into Warm-Up Mode. The Sensor Unit will enter Zero Calibration Mode from Zero Calibration Fault when the user initiates a Zero Calibration. The Sensor Unit will leave Span Calibration Fault and enter Span and Zero Calibration Mode when the user initiates the Span and Zero Calibration. The Sensor Unit will enter Diagnostic Mode from Diagnostic Fault when the user initiates the diagnostic.

4.1.2 Area Monitor Top Level Software

The Area Monitor software will be modeled as a finite state machine, implementing the Area Monitor functions of the system state machine given in Figure 4-1. Figure 4-3 shows the state diagram of and the transitions between the top-level states of the Area Monitor software. The primary purpose of the Area Monitor is to control the alarms and relays, and to control the user interfaces. The Area Monitor will act as an interface between the user and the Sensor Unit and the remote computer.

The user can perform a Configuration, Zero Calibration, Span and Zero Calibration, and a Diagnostic from both the remote computer and the built-in interface. The Area Monitor will have separate top-level states for each operation, one for the remote computer initiated operation and one for the built-in interface initiated operation.

The Area Monitor must request and update the PPM every second, and must provide the user interface to initiate any operation. The Area Monitor will normally operate in Normal Operation Mode. The Area Monitor will remain in Normal Operation Mode if the alarm status is no-alarm, and will transition into Alarm Mode when the alarm status is low, high, or over-range.

The Area Monitor will transition to the operation state depending on the operation selected and the interface being used.

Alarm Mode is entered when the Sensor Unit has detected an elevated PPM. The Area Monitor will remain in Alarm Mode as long as the alarm status is low, high, or over-range. Normal Operation Mode is entered when the alarm status is no-alarm. The Area Monitor will only allow the user to initiate a configuration from Alarm Mode. The Area Monitor will transition into one of the two Configuration Modes when the user selects the configuration option. Which one of the two Configuration Mode states entered is determined by where the user initiated the operation.

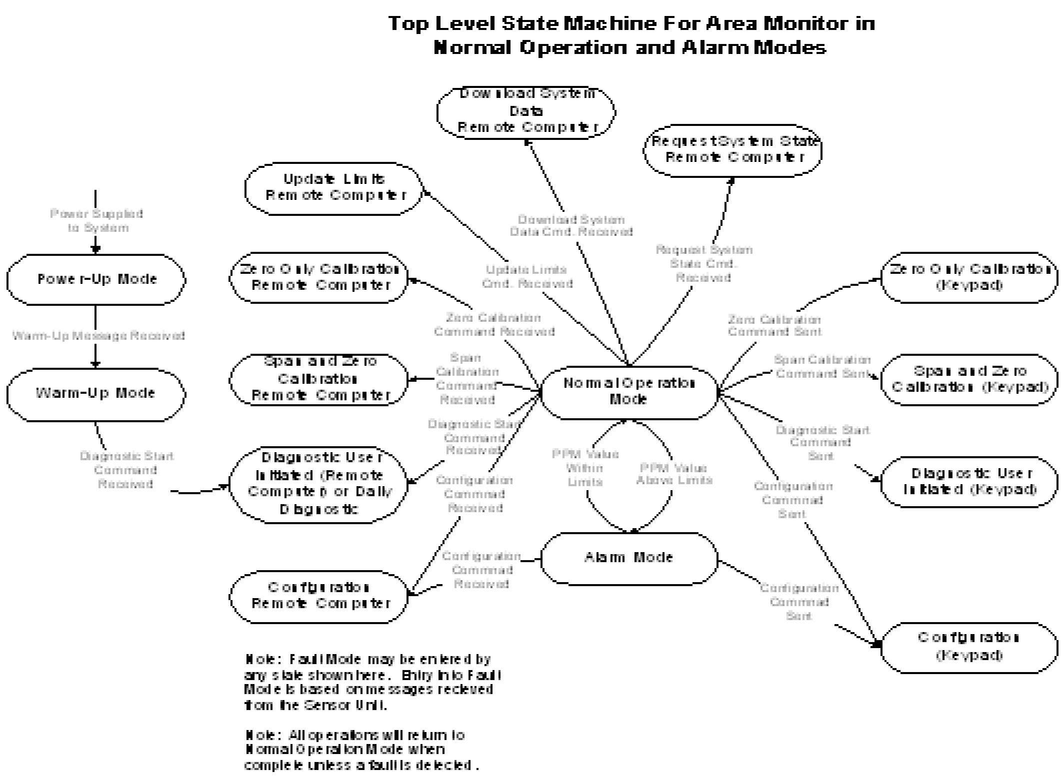

Figure 4-3: Top-level state diagram of the Area Monitor in Normal Operation or Alarm Mode.

The Sensor Unit will send fault messages to the Area Monitor when a fault is detected. The Area Monitor will transition into Fault Mode after receiving a fault indication message. The Area Monitor will leave Fault Mode to clear the current outstanding fault as in accordance with the requirements listed in Section 3.1.12. When a Loss of Power Fault is cleared, the system must synchronize before entering Warm-Up Mode after correcting the fault. The Area Monitor will wait for the Sensor Unit to acknowledge the loss of power acknowledge message, and will then acknowledge the acknowledgement, to synchronize the entry into Warm-Up Mode.

The Area Monitor will enter Power-Up Mode when power is first supplied to the system. Warm-Up Mode is entered from Power-Up Mode after receiving the warm-up message from the Sensor Unit.

The Area Monitor enters Warm-Up Mode from Power-Up Mode. The Area Monitor sends the acknowledge message to the Sensor Unit, informing the Sensor Unit that the Area Monitor is currently in Warm-Up Mode. This prevents the Sensor Unit from sending the sensor data to the Area Monitor before the Area Monitor is ready to receive the data. The Area Monitor will leave Warm-Up Mode and enter Diagnostic Mode after receiving the diagnostic command from the Sensor Unit.

The Area Monitor will enter Diagnostic Mode after receiving the diagnostic command from the Sensor Unit in Warm-Up Mode, or when the user selects the diagnostic option from the user interface. The Area Monitor will send the acknowledge message to the Sensor Unit after entering Diagnostic Mode. This prevents the Area Monitor from missing the diagnostic test command, sent by the Sensor Unit to test the RS-232 communications link between the Sensor Unit and the Area Monitor. The Area Monitor will send the acknowledge command after receiving the diagnostic report command, preventing the Sensor Unit from sending the diagnostic report before the Area Monitor is ready to accept the diagnostic report. The Area Monitor will acknowledge the receipt of the diagnostic report, sending the acknowledge message to the Sensor Unit, to prevent the Sensor Unit from entering Normal Operation Mode before the Area Monitor.

Configuration Mode is entered from Normal Operation or Alarm Mode, when the user selects the configuration option. After sending the configuration command to the Sensor Unit, the Area Monitor will wait until the Sensor Unit acknowledges the configuration command. This is to prevent the Area Monitor from sending the configuration data before the Sensor Unit is ready to accept it. The Area Monitor and Sensor Unit must enter Normal Operation Mode at the same time. The Area Monitor will send the acknowledge command to the Sensor Unit after receiving the configuration complete ok message from the Sensor Unit. This will allow the entry into Normal Operation Mode to be synchronized.

Zero Calibration Mode is entered from Normal Operation Mode in response to the user selecting the zero calibration option from the man-machine interface. The Area Monitor will wait for the Sensor Unit to enter Zero Calibration Mode, to prevent the system from losing synchronization. The Area Monitor will wait for the Sensor Unit to acknowledge the zero calibration command before performing the rest of the zero calibration routine. The Area Monitor will acknowledge the receipt of the zero calibration complete ok message, to prevent the Sensor Unit and Area Monitor from entering Normal Operation Mode at different times and causing the system to lose synchronization.

The Span and Zero Calibration Mode is entered from Normal Operation Mode when the user selects the span and zero calibration operation from the user interface. The Sensor Unit and Area Monitor must synchronize before performing the Span and Zero Calibration. The Area

Monitor will wait for the Sensor Unit to acknowledge the entry into Span and Zero Calibration Mode before performing the span and zero calibration routine. The Area Monitor will acknowledge the zero calibration complete and span calibration complete messages to prevent the Sensor Unit software from moving too far ahead of the Area Monitor software.

4.2 DESIGN OF THE TOP-LEVEL STATE MACHINE

The preceding sections describe the design of the state machine for the two parts of the distributed system. The software will maintain state variables to maintain the current state of all state machines in the system. These variables control which state is executed each time through the top-level loop. The state variables of those operations having internal state machines are reset when the operation is initialized. The top-level loop of the software determines which function to execute based on the top-level state. The design of the top-level loop for the Sensor Unit is presented first, followed by the design of the top-level loop for the Area Monitor. Both loops are infinite loops and never exit unless power is lost.

Figure 4-4 shows the pseudocode for the top-level software loop in the Sensor Unit. The Sensor Unit will check to verify if the Sensor Module is present every time through the main loop. If the Sensor is not present, Fault Mode is entered and a Sensor Module Missing Fault is issued. If the Sensor Module is present, the Sensor Module Detected flag is set, informing the Fault Mode routine that the Sensor Module is present, if there is a Sensor Module Missing Fault outstanding. The top-level state then determines what code is executed.

```
WHILE( TRUE )
        IF( Sensor Module There Pin == 0 ) THEN
          Sensor Module Detected = TRUE
        ELSE
          Sensor Module Detected = FALSE
          Enter Fault Mode and Issue a Sensor Module Missing Fault
        END IF

        IF( Top-Level State == Normal Operation Mode ) THEN
          Perform Normal Operation Mode Function
        ELSE IF( Top-Level State == Alarm Mode ) THEN
          Perform Alarm Mode Function
        ELSE IF( Top-Level State == Fault Mode ) THEN
          Perform Fault Mode Function
        ELSE IF( Top-Level State == Diagnostic Mode ) THEN
          Perform Diagnostic Mode Function
        ELSE IF( Top-Level State == Configuration Mode ) THEN
          Perform Configuration Operation
        ELSE IF( Top-Level State == Zero Calibration Mode ) THEN
          Perform Zero Calibration Function
        ELSE IF( Top-Level State == Span and Zero Calibration ) THEN
          Perform Span and Zero Calibration
        ELSE IF( Top-Level State == Sensor Warm-Up Mode ) THEN
          Perform Sensor Warm-Up Function
        ELSE IF( Top-Level State == Power-Up Mode ) THEN
          Perform Power-Up Mode Function
        ELSE IF( Top-Level State == Download System Data ) THEN
          Perform Download System Data
        ELSE IF( Top-Level State == Update Limits Mode ) THEN
          Perform Update Limits Function
        END IF-ELSE

END WHILE
```

Figure 4-4: Pseudocode of the top-level software loop in the Sensor Unit.

Figure 4-5 shows the pseudocode for the top-level software loop in the Area Monitor. The Area Monitor performs the Request System State operation if the Send Synchronize Response flag is set. The flag is tested every time through the main loop. The code then executed depends on the top-level state.

```
WHILE( TRUE )

        IF( Send Synchronize Response == TRUE ) THEN
          Send Synchronize Response = FALSE
          Perform Request System State Function
        END IF

        IF( Top-Level State == Normal Operation Mode ) THEN
          Perform Normal Operation Mode Function
        ELSE IF( Top-Level State == Alarm Mode ) THEN
          Perform Alarm Mode Function
        ELSE IF( Top-Level State == Fault Mode ) THEN
          Perform Fault Mode Function
        ELSE IF( Top-Level State == Diagnostic Mode ) THEN
          Display Diagnostic Message to User
          Perform Diagnostic Mode Function
        ELSE IF( Top-Level State == Keypad Configuration Mode ) THEN
          Perform Keypad Configuration Operation
        ELSE IF( Top-Level State == Keypad Zero Calibration Mode ) THEN
          Perform Keypad Zero Calibration Function
        ELSE IF( Top-Level State == Keypad Span and Zero Calibration ) THEN
          Perform Keypad Span and Zero Calibration
        ELSE IF( Top-Level State == Remote Computer Configuration Mode ) THEN
          Display Configuration Message to User
          Perform Remote Computer Configuration Operation
        ELSE IF( Top-Level State == Remote Computer Zero Calibration Mode ) THEN
          Display Zero Calibration Message to User
          Perform Remote Computer Zero Calibration Function
        ELSE IF( Top-Level State == Remote Computer Span and Zero Calibration ) THEN
          Display Span and Zero Calibration Message to User
          Perform Remote Computer Span and Zero Calibration
        ELSE IF( Top-Level State == Sensor Warm-Up Mode ) THEN
          Display Sensor Warm-Up Message to User
          Perform Sensor Warm-Up Function
        ELSE IF( Top-Level State == Power-Up Mode ) THEN
          Display Power-Up Mode Message to User
          Perform Power-Up Mode Function
        ELSE IF( Top-Level State == Download System Data ) THEN
          Display Download System Data Message to User
          Perform Download System Data
        ELSE IF( Top-Level State == Update Limits Mode ) THEN
          Display Update Limits Message to User
          Perform Update Limits Function
        END IF-ELSE
END WHILE
```

Figure 4-5: Pseudocode for the top-level software loop in the Area Monitor.

4.3 DESIGN OF AREA MONITOR TO SENSOR UNIT RS-232 ISR IN THE SENSOR UNIT

The Sensor Unit will use an interrupt to handle messages received from the Area Monitor over the RS-232 communications link. The ISR is executed when the built-in UART receives a message. The current top-level state determines what action the ISR takes in response to the message.

The ISR will receive commands and data received from the Area Monitor. The ISR causes transitions in response to commands received. The ISR will collect data sent from the Area Monitor and will inform the main software when all data has been received.

Pseudocode for the Sensor Unit RS-232 ISR is shown in Figure 4-6. The received byte is stored in the Message global variable. All operations can be initiated from Normal Operation Mode. In Normal Operation Mode, the message is a command to initiate an operation. The ISR sets the top-level state causing the transition into the requested routine. Several routines have internal state machines and internal variables that must be set to the initial values. Any inner state variables and internal variables used during the requested routine in the ISR. If the top-level state is Alarm Mode, only a configuration may be initiated. Any other command will be ignored.

The Area Monitor Message Received flag is set to TRUE if the Sensor Unit is currently in Diagnostic Mode, Zero Calibration Mode, or Fault Mode. The software for these three modes will take action based on the message. Configuration and Update Limits Mode will receive data from the Area Monitor. The ISR collects this data in the temporary buffer. After receiving all configuration data or limits data, the All Configuration Data flag or the All Limits Data Received flag are set to TRUE respectively.

During a Span and Zero Calibration, the Area Monitor sends commands and data to the Sensor Unit. The ISR collects the scale factor and reference PPM if the Span Calibration State is Receiving Data. The Area Monitor Message Received flag is set to TRUE in all other span calibration states. After the last byte of span calibration data is received, the ISR sets the All

Span Calibration Received flag to TRUE. The number of data bytes sent to and from the Area Monitor is constant.

```
Message = Byte Received

IF ( Top-Level State == Normal Operation Mode ) THEN
        IF( Message == Request PPM Command ) THEN
                Send PPM = TRUE
        ELSE IF( Message == Zero Calibration Command ) THEN
                Top-Level State = Zero Only Calibration
                Time Counter = 0
        ELSE IF( Message == Span Calibration Command ) THEN
                Top-Level State = Span and Zero Calibration
                Time Counter = 0
                Write Address = Number of Bytes In Scale Factor and Reference PPM
                Span Calibration State = Wait For Zero Calibration Start Command
        ELSE IF( Message == Diagnostic Start Command ) THEN
                Top-Level State = Diagnostic  User Initiated
        ELSE IF( Message == Download To Remote PC Command ) THEN
                Top-Level State = Download To Remote PC
                Diagnostic State = Sensor Diagnostic
                Time Counter = 0
        ELSE IF( Message == Download System Data Command ) THEN
                Top-Level State = Download System Data
        ELSE IF( Message == Configuration Command ) THEN
                Top-Level State = Configuration
                Write Address = Size of Configuration Table
                Time Counter = 0
        ELSE IF( Message == Update Limits Command ) THEN
                Top-Level State = Update Limits
                Write Address = 0
                Time Counter = 0
        END IF-ELSE
ELSE IF( Top-Level State == Alarm Mode ) THEN
        IF( Message == Configuration Command ) THEN
                Top-Level State = Configuration
                Write Address = Size of Configuration Table
                Time Counter = 0
        END IF
ELSE IF( Top-Level State == Diagnostic ) THEN
        Area Monitor Message Received = TRUE
ELSE IF( Top-Level State == Configuration ) THEN

        IF( Write Address > 0 ) THEN
                Store Message in Temporary Buffer at Configuration Table Size - Write Address
                Write Address = Write Address - 1
        END IF

        IF( Write Address == 0 ) THEN
                All Configuration Data Received = TRUE
        END IF
ELSE IF( Top-Level State == Zero Only Calibration ) THEN
        Area Monitor Message Received = TRUE
ELSE IF( Top-Level State == Span and Zero Calibration ) THEN
        IF( Span Calibration State == Receiving Data ) THEN
                IF( Write Address > 0 ) THEN
                        Store Message in Temporary Buffer at Span Bytes Received - Write Address
                        Write Address = Write Address - 1
                END IF

                IF( Write Address == 0 ) THEN
                     All Span Calibration Data Received = TRUE
                END IF
        ELSE
              Area Monitor Message Received = TRUE
        END IF
ELSE IF( Top-Level State == Fault Mode ) THEN
        Area Monitor Message Received = TRUE
ELSE IF( Top-Level State == Update Limits ) THEN
        IF( Write Address < Size of Limits Table ) THEN
                Store Message in Temporary Buffer at Write Address
                Write Address = Write Address + 1
        END IF

        IF( Write Address == Size of Limits Table ) THEN
                All Limits Data Received = TRUE
        END IF
END IF-ELSE
```

Figure 4-6: Pseudocode for the Sensor Unit RS-232 ISR handling messages from the Area Monitor.

4.4 DESIGN OF SENSOR UNIT TO AREA MONITOR RS-232 ISR IN THE AREA MONITOR

The Area Monitor will receive messages and data from the Sensor Unit via the RS-232 communications link. The Area Monitor RS-232 ISR will receive and handle bytes sent from the Sensor Unit. The Area Monitor RS-232 ISR will execute when a byte is received from the Sensor Unit.

The byte will be handled based on the top-level state of the Area Monitor. Messages sent from the Sensor Unit are data in response to a request from the Area Monitor, fault messages, the warm-up message, sensor data, or the diagnostic start command. The top-level state determines how the ISR will handle the message. The ISR causes a state transition in response to a command. Data sent from the Sensor Unit during Sensor Warm-Up or Download System Data is collected and a flag set after the last byte of data is received.

Figure 4-7 shows the pseudocode for the Area Monitor RS-232 ISR. The received byte is stored in the Message global variable, and action is taken based on the top-level state. If the received byte is the Timeout message and data is not being sent from the Sensor Unit (this prevents a byte of data identical to the timeout message from triggering a false timeout) the top-level state is set to Normal Operation Mode. Data is received from the Sensor Unit in the Sensor Warm-Up and Download System Data Modes.

In Normal Operation Mode and Alarm Mode the alarm status and PPM is received from the Sensor Unit in response to the PPM request, otherwise the received byte is checked to see if it is a fault message. If the byte is a fault message Fault Mode is entered.

```
Message = Byte Received

IF( (Message == Timeout Message) AND (Top-Level State NOT Warm-Up or Download System Data) ) THEN
        Top-Level State = Normal Operation
        Send Timeout Message to Remote Computer
ELSE IF( Top-Level State == Normal Operation Mode ) THEN
        IF( Send PPM ) THEN
                IF( Write Address > 0 ) THEN
                        Store Message in Temporary Buffer at 6 - Write Address
                        Write Address = Write Address - 1
                        IF( Write Address == 0 ) THEN
                                Check For Fault Message
                                Get Status and PPM from Temporary Buffer
                                PPM Received = TRUE
                        END IF
                END IF
        ELSE
                Check For Fault Message
        END IF-ELSE
ELSE IF( Top-Level State == Alarm Mode ) THEN
        IF( Send PPM ) THEN
                IF( Write Address > 0 ) THEN
                        Store Message in Temporary Buffer at 6 - Write Address
                        Write Address = Write Address - 1
                        IF( Write Address == 0 ) THEN
                                Check For Fault Message
                                Get Status and PPM from Temporary Buffer
                                PPM Received = TRUE
                        END IF
                END IF
        ELSE
                Check For Fault Message
        END IF-ELSE
ELSE IF( Top-Level State == Keypad or Remote Computer Configuration ) THEN
        Check For Fault Message
        Sensor Unit Message Received = TRUE
ELSE IF( Top-Level State == Keypad or Remote Computer Zero Calibration ) THEN
        Check For Fault Message
        Sensor Unit Message Received = TRUE
ELSE IF( Top-Level State == Keypad or Remote Computer Zero and Span Calibration ) THEN
        Check For Fault Message
        Sensor Unit Message Received = TRUE
ELSE IF( Top-Level State == Diagnostic Mode ) THEN
        IF( Diagnostic State == Wait For Diagnostic Report Data ) THEN
                Sensor Unit Message Received = TRUE
        ELSE
                Check For Fault Message
                Sensor Unit Message Received = TRUE
        END IF-ELSE
ELSE IF( Top-Level State == Sensor Warm-Up ) THEN
        IF( Warm-Up State == Receive New Sensor Data ) THEN
                IF( Write Address < Warm-Up Data Size )
                        Store Message in Temporary Buffer at Write Address
                        Write Address = Write Address + 1
                        IF( Write Address == Warm-Up Data Size ) THEN
                                Set All Data Received Flag
                        END IF
                END IF
        ELSE IF( Wait for Diagnostic Command ) THEN
                Check For Fault Message
                Sensor Unit Message Received = TRUE
        END IF-ELSE
ELSE IF( Top-Level State == Download System Data ) THEN
        IF( Write Address < Size Download Data ) THEN
                Store Message in Temporary Buffer at Write Address
                Write Address = Write Address + 1
                IF( Write Address == Size Download Data ) THEN
                        Set All Download Data Received Flag
                END IF
        END IF
ELSE IF( Top-Level State == Fault Mode ) THEN
        IF( Message == Warm-Up Message ) THEN
                Top-Level State = Warm-Up Mode
                Warm-Up State = Receive New Sensor Data
                Write Address = 0
                Send Warm-Up Message to Remote PC = TRUE
        END IF
ELSE IF( Top-Level State == Power-Up Mode ) THEN
        IF( Message == Warm-Up Message ) THEN
                Top-Level State = Warm-Up Mode
                Warm-Up State = Receive New Sensor Data
                Write Address = 0
                Send Warm-Up Message to Remote PC = TRUE
        ELSE IF( Message == Loss of Power Fault Message ) THEN
                Top-Level State = Fault Mode
                Outstanding Fault = Loss of Power Fault
        ELSE IF( Message == Sensor Module Missing ) THEN
                Top-Level State = Fault Mode
                Outstanding Fault = Sensor Module Missing Fault
        ELSE IF( Message == Sensor Module Past Specified Lifetime Fault Message ) THEN
                Top-Level State = Fault Mode
                Outstanding Fault = Sensor Module Past Specified Lifetime Fault
        END IF-ELSE
ELSE IF( Top-Level State == Update Limits ) THEN
        Sensor Unit Message Received = TRUE
END IF-ELSE
```

Figure 4-7: Pseudocode for the ISR in the Area Monitor to handle incoming messages from the Sensor Unit.

Configuration, Zero Calibration, and Span and Zero Calibration will only receive the respective operation complete message or a fault message from the Sensor Unit. If a fault message is received the Area Monitor enters Fault Mode, otherwise the Sensor Unit Message Received flag is set. If the Area Monitor is in Diagnostic Mode and waiting for the diagnostic report, the Sensor Unit Message Received flag is set, otherwise the message is checked for a fault message and the Sensor Unit Message Received flag is set. If the top-level state is Download System Data, the ISR stores each byte received from the Sensor Unit in the temporary buffer and set the All Download Data Received flag after receiving the last byte of download data.

If the top-level state is Sensor Warm-Up Mode, the Area Monitor will receive the new sensor data, the diagnostic command, or a fault message. If waiting for the new sensor data, then the received bytes are stored in the temporary buffer, and the All Data Received flag is set after receiving the last byte. If waiting for the diagnostic command, the Area Monitor enters Diagnostic Mode after receiving the diagnostic command, and enters Fault Mode if a fault message is received.

During Fault Mode, the Area Monitor may receive the warm-up message from the Sensor Unit indicating the Sensor Module has been inserted. Warm-Up Mode is entered when the ISR receives the warm-up message when in Fault Mode.

In Power-Up Mode, the Sensor Unit checks for the presence of the Sensor Module and for a Loss of Power Fault. In Power-Up Mode, the ISR can receive either the warm-up message or a fault message. If the warm-up message is received, the ISR causes the Area Monitor to enter Sensor Warm-Up Mode, and if a fault message is received, the ISR causes the transition into Fault Mode.

4.5 DESIGN OF TIMER ISR

The timer interrupt will be used to track durations of time. The Sensor Unit must track time to determine if the communications diagnostic has failed, and to sample the sensor output during Sensor Warm-Up or a calibration. The Area Monitor tracks time to request the PPM once per second, and to initiate the daily diagnostic. The system must also provide a means to exit an operation that is taking too long. The timer interrupt will to determine when an operation has timed out. Both the Sensor Unit and the Area Monitor will track time using the built-in timer interrupt. The design of the timer ISR for the Sensor Unit is presented first, followed by the design of the timer interrupt for the Area Monitor.

The Sensor Unit must track time to take samples every second during a calibration, or Sensor Warm-Up, and to timeout when an operation takes too long. The timeout will result if the operation takes longer than the maximum specified time. This prevents the system from waiting forever to finish an operation. When an operation times out, the Sensor Unit sends the Timeout Message to the Area Monitor and enters Normal Operation Mode. The built-in Sensor Unit timer interrupt is set to interrupt approximately every tenth of a second.

Figure 4-8 shows the pseudocode for the Sensor Unit timer ISR. The actions taken depend on the top-level state of the system. If the top-level state is Update Limits, or Configuration the timeout test is performed every time the timer ISR executes. The time for sample flag is set every second in Warm-Up Mode. The ISR sets the time for sample flag every second and checks for a timeout when the top-level state is Zero Calibration or Span and Zero Calibration. If the top-level state is Diagnostic Mode, the ISR will check for the timeout of the communications diagnostic. If the communications diagnostic times out a diagnostic fault has occurred and the Sensor Unit enters Fault Mode, issuing a diagnostic fault. If a timeout occurs, the timeout message is sent to the Area Monitor and the Sensor Unit enters Normal Operation Mode.

```
Time Counter = Time Counter + 1

IF( Top-Level State == Diagnostic ) THEN
        IF( Diagnostic State == Wait For Area Monitor Response ) THEN
                IF( Time Counter >= Maximum Timer Interrupts to Wait for Area Monitor Response ) THEN
                        Top-Level State = Fault Mode
                        Outstanding Fault = Diagnostic Fault
                        Time Counter = 0
                END IF
        END IF
ELSE IF( Top-Level State == Configuration ) THEN
        IF( Time Counter >= Maximum Timer Interrupts to Wait for Configuration to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Area Monitor
                Time Counter = 0
        END IF
ELSE IF( Top-Level State == Zero Only Calibration ) THEN
        IF( Time Counter % Number of Timer Interrupts Per Second == 0 ) THEN
                Time For Sample = TRUE
        END IF

        IF( Time Counter >= Maximum Timer Interrupts to Wait for Zero Calibration to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Area Monitor
                Time Counter = 0
        END IF
ELSE IF( Top-Level State == Span and Zero Calibration ) THEN
        IF( Time Counter % Number of Timer Interrupts Per Second == 0 ) THEN
                Time For Sample = TRUE
        END IF

        IF( Time Counter >= Maximum Timer Interrupts to Wait for Span and Zero Calibration to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Area Monitor
                Time Counter = 0
        END IF
ELSE IF( Top-Level State == Warm-Up Mode ) THEN
        IF( Time Counter % Number of Timer Interrupts Per Second == 0 ) THEN
                Time For Sample = TRUE
        END IF
ELSE IF( Top-Level State == Update Limits ) THEN
        IF( Time Counter >= Maximum Timer Interrupts to Wait for Update Limits to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Area Monitor
                Time Counter = 0
        END IF
END IF-ELSE
```

Figure 4-8: Pseudocode for the Sensor Unit timer ISR.

The Area Monitor must send the PPM request every second and must timeout when waiting for user input. The built-in timer is set to interrupt approximately every tenth of a second. Figure 4-9 shows the pseudocode for the timer ISR in the Area Monitor. The top-level state determines the action taken. If the Area Monitor is in Normal Operation or Alarm Mode, the new PPM sample must be requested every second. The timer ISR sets the Time for PPM flag every second. The Area Monitor must prevent a Configuration or Update Limits operation from waiting forever for the user to enter required data. The timer ISR times out if, either operation takes longer than the maximum allowed time. When a timeout is detected, the timer ISR sends the Timeout Message to the remote computer and enters Normal Operation Mode.

```
Time Counter = Time Counter + 1
Daily Diagnostic Counter = Daily Diagnostic Counter + 1

IF( Top-Level State == Normal Operation Mode ) THEN
        IF( Time Counter % Number of Timer Interrupts Per Second == 0 ) THEN
                Time For PPM = TRUE
        END IF
ELSE IF( Top-Level State == Alarm Mode ) THEN
        IF( Time Counter % Number of Timer Interrupts Per Second == 0 ) THEN
                Time For PPM = TRUE
        END IF
ELSE IF( Top-Level State == Keypad or Remote Computer Configuration ) THEN
        IF( Time Counter >= Maximum Timer Interrupts to Wait for Configuration to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Remote Computer
                Time Counter = 0
        END IF
ELSE IF( Top-Level State == Update Limits ) THEN
        IF( Time Counter >= Maximum Timer Interrupts to Wait for Update Limits to Complete ) THEN
                Top-Level State = Normal Operation Mode
                Send Time Out Message to Remote Computer
                Time Counter = 0
        END IF
END IF-ELSE

IF( Daily Diagnostic Counter >= Number Timer Interrupts in 24 Hours ) THEN
        IF( Top-Level State == Normal Operation Mode ) THEN
                Daily Diagnostic Flag = TRUE
        END IF

        Daily Diagnostic Counter = 0
END IF
```

Figure 4-9: Pseudocode for the Area Monitor timer ISR.

4.6 DESIGN OF AREA MONITOR/REMOTE PC RS-232 ISR

The remote computer will send commands and data to the Area Monitor. The remote computer RS-232 ISR receives and handles messages from the remote computer. The messages received from the remote computer are commands to initiate an operation, or data sent during an operation. The ISR will change the top-level state in response to commands and informs the main software after all data is received. This ISR executes when a message is received from the remote computer.

Figure 4-10 shows the pseudocode for the remote computer RS-232 ISR. The ISR determines what action to take based on the top-level state and the message. The Request System State command is responded to unless the remote computer is sending a data set.

In Normal Operation Mode, the remote computer can initiate any operation if the Area Monitor is not waiting to receive the alarm status and PPM. This prevents the Area Monitor from switching states and subsequently interpreting part of the PPM as a message. The ISR causes the transition to the appropriate state to perform the requested operation. The remote computer can only initiate a configuration from Alarm Mode. The ISR causes the transition to Remote Computer Configuration Mode to occur in response to the configuration command, and takes no action on any request other than the request system state command.

In Remote Computer Configuration Mode, the ISR collects the new configuration data sent from the remote computer in the temporary buffer, setting the All Configuration Data Received flag after receiving the last byte. When the top-level state is Remote Computer Zero Configuration the ISR will set the Remote Computer Message Received flag, unless the message is the request system state command.

```
R_Message = Read Received Byte from External UART

IF( Top-Level State == Normal Operation Mode ) THEN
        IF( (R_Message == Diagnostic Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Diagnostic Mode
                Diagnostic State = Wait For Area Monitor Test Command
                Send Diagnostic Command to Sensor Unit
        ELSE IF( (R_Message == Zero Calibration Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Zero Calibration Mode
                Zero Calibration State = Wait For Zero Calibration Start Command
                Send Zero Calibration Command to Sensor Unit
        ELSE IF( (R_Message == Span and Zero Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Span and Zero Calibration Mode
                Span Calibration State = Wait For Zero Calibration Start Command
                Send Span and Zero Calibration Command to Sensor Unit
        ELSE IF( (R_Message == Download System Data Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Download System Data Mode
                Write Address = 0
                Send Download System Data Command to Sensor Unit
        ELSE IF( (R_Message == Configuration Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Remote Computer Configuration Mode
                Configuration State = Wait For Remote Computer Data
                Write Address = 0
                Time Counter = 0
        ELSE IF( (R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        ELSE IF( (R_Message == Update Limits Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Update Limits Mode
                Write Address = 0
                Time Counter = 0
        END IF-ELSE
ELSE IF( Top-Level State == Alarm Mode ) THEN
        IF( (R_Message == Configuration Command) AND ( Send PPM Flag == FALSE) ) THEN
                Top-Level State = Remote Computer Configuration Mode
                Configuration State = Wait For Remote Computer Data
                Write Address = 0
                Time Counter = 0
        ELSE IF( (R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF-ELSE
ELSE IF( Top-Level State == Remote Computer Configuration Mode ) THEN
        IF( Write Address < Configuration Table Size ) THEN
                Store Message in Temporary Buffer at Write Address
                Write Address = Write Address + 1
        ELSE IF( (R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF-ELSE

        IF( Write Address == Size Configuration Table ) THEN
                All Configuration Data Received = TRUE
        END IF
ELSE IF( Top-Level State == Remote Computer Zero Calibration ) THEN
        IF( R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        ELSE
                Remote Computer Message Received = TRUE
        END IF-ELSE
ELSE IF( Top-Level State == Remote Computer Span and Zero Calibration ) THEN
        IF( SCalState == Receive Data ) THEN
                IF( Write Address < Size Span Calibration Data ) THEN
                        Store Message in Temporary Buffer at Write Address
                        Write Address = Write Address + 1
                END IF

                IF( Write Address == Size Span Calibration Data ) THEN
                        All Span Calibration Data Received = TRUE
                END IF
        ELSE IF( R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        ELSE
                Remote Computer Message Received = TRUE
        END IF-ELSE
ELSE IF( Top-Level State == Sensor Warm-Up Mode ) THEN
        IF( R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF
ELSE IF( Top-Level State == Download System Data Mode ) THEN
        IF( R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF
ELSE IF( Top-Level State == Update Limits Mode ) THEN
        IF( Write Address < Limits Table Size ) THEN
                Store Message in Temporary Buffer at Write Address
                Write Address = Write Address + 1
        ELSE IF( (R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF-ELSE

        IF( Write Address == Size Limits Table ) THEN
                All Limits Data Received = TRUE
        END IF
ELSE IF( Top-Level State == Fault Mode ) THEN
        IF( R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        ELSE

                Remote Computer Message Received = TRUE
        END IF
ELSE
        IF( (R_Message == Request System State) ) THEN
                Send Synchronize Response = TRUE
        END IF
END IF-ELSE
```

Figure 4-10: Pseudocode of the incoming RS-232 Remote Computer ISR.

In Remote Computer Span and Zero Calibration Mode, the ISR will set the Remote Computer Message Received flag, unless the message is the request system state command or the Span Calibration State is Receiving Data. When the Span Calibration State is Receiving Data, the ISR will collect the span calibration data, and will set the All Span Calibration Data Received flag after receiving the last byte of span data.

The ISR will only respond to the request system state command in Sensor Warm-Up and Download System Data Modes. In Fault Mode, the ISR will set the Remote Computer Message Received flag, unless the message is the request system state command, the Fault Mode software will handle the message. In Update Limits Mode, the ISR collects the limits data from the remote computer and set the All Limits Data Received flag after receiving the last byte of limits data.

An external UART chip provides the RS-232 communications link between the Area Monitor and Remote Computer. To write to the registers in the external UART, the write operation is triggered and the address signals are set to the desired register. The data is then provided and the data write is triggered, writing the data to the selected register. To read from one of the registers, the read operation is triggered, and the address signals select the desired register to read from. The software waits 150ns and then reads the data from the UART chip. To send to the remote computer the software waits 5us to allow the previous transmit to complete, then the transmit register is written to with the byte to send. Reading the receive register, retrieves the last character received by the UART chip.

4.7 DESIGN OF KEYPAD ISR

The Area Monitor must respond to input from the 5-button keypad. The user to move through the menus and to enter data during a Configuration or a Span and Zero Calibration uses the keypad. The keypad ISR is executed when a key press is detected and will handle keypad input. When a key is pressed the keypad input for the pressed key is pulled low, the input is high if the key is not pressed. The inputs of all five keys are ANDed together, generating the keypad input detected signal providing an interrupt signal when any key is pressed. When the keypad input detected signal makes a high to a low transition, the keypad ISR is executed, a low to high transition is ignored.

Figure 4-11 shows the pseudocode of the keypad ISR. The keypad ISR will determine which key was pressed by testing the status of the input for each of the five keys. The KeyPress variable is a global variable containing the last key pressed. The main software to take action based on keypad input uses this variable.

The keypad ISR also handles updating the user-entered data. The Area Monitor will display the current value to the user and the user will alter that value using the keypad. The value displayed is maintained in a global floating-point variable. Only one of the three digits displayed can be incremented or decremented at a time, the fourth displayed character is the decimal point. The keypad ISR also maintains the digit that is currently being edited in a global variable. When the user presses the up key, the digit currently being edited is incremented by 1 and when the down key is pressed the digit is decremented by 1. If the incremented digit rolls over from 9 to 0, the digit immediately to the left is incremented by 1, and if the decremented digit rolls over from 0 to 9, the digit immediately to the left is decremented by 1. The left key moves the Digit Currently Being Edited one place to the left and the right key moves Digit

Currently Being Edited one place to the right. If the user moves the currently editing digit to the position of the decimal point, the currently editing position is moved one more place in the direction chosen by the user. The Enter key accepts the value, setting the KeypadDataReady flag to TRUE.

```
IF( Keypad Input Detected ) THEN
        IF( Up Key Pressed ) THEN
                KeyPress = Up Key
                Increment Data based on the Digit Currently Being Edited
        ELSE IF( Down Key Pressed ) THEN
                KeyPress = Down Key
                Decrement Data based on the Digit Currently Being Edited
        ELSE IF( Right Key Pressed ) THEN
                KeyPress = Right Key
                Digit Currently Being Edited = Digit Currently Being Edited + 1

                IF(Digit Currently Being Edited == Position of Decimal Point ) THEN
                        Digit Currently Being Edited = Digit Currently Being Edited + 1
                END IF
        ELSE IF( Left Key Pressed ) THEN
                KeyPress = Left Key
                Digit Currently Being Edited = Digit Currently Being Edited - 1

                IF(Digit Currently Being Edited == Position of Decimal Point ) THEN
                        Digit Currently Being Edited = Digit Currently Being Edited - 1
                END IF
        ELSE IF( Enter Key Pressed ) THEN
                KeyPress = Enter Key
                KeypadDataReady = TRUE
        END IF-ELSE
END IF
```

Figure 4-11: Pseudocode for the Area Monitor keypad ISR.

4.8 DESIGN OF ALARM SILENCE BUTTON ISR

An alarm silence button is provided to turn off the alarms before the PPM returns to a safe level. The latching and acknowledge settings determine the actions required to deactivate the alarm horns and relays. The latching settings determine if relays can be reset after leaving the low or high alarm states. The acknowledge settings determine if the relays can be reset by the alarm silence button. There are two latching and acknowledge settings, one for the low alarm relay and the second for the high alarm relay.

The alarm silence button ISR is executed when the alarm silence button is pressed. Figure 4-12 shows the pseudocode for the alarm silence ISR. The alarm horns are always turned off when the alarm silence button is pressed. The actions taken with respect to the relays are determined based on the alarm state, the top-level state, and the acknowledge settings. If the top-level state is Normal Operation Mode, both relays are reset regardless of the latching and acknowledge settings. If the top-level state is Alarm Mode, the alarm state and the acknowledge values determine the actions to take with respect to the relays. For the relay to be reset when the button is pressed the acknowledge setting must be OFF, otherwise the relay is not reset. When the alarm state is no-alarm, both relays may be turned off depending on the low and high acknowledge settings. When the alarm state is low alarm, only the low relay can be reset, and when in high alarm, only the high relay can be reset.

```
Turn Off Low and High Alarm Horns


IF (Top-Level State == Alarm Mode ) THEN
        IF (Alarm Status == No-Alarm ) THEN
                Reset Low Relay
                Reset High Relay
                Low Relay Reset By Button = FALSE
                High Relay Reset By Button = FALSE
        ELSE IF (Alarm Status == Low Alarm ) THEN
                IF (Low Acknowledge Flag Off) THEN
                        Reset Low Relay
                        Low Relay Reset By Button = FALSE
                END IF
        ELSE IF (Alarm Status == High Alarm ) THEN
                IF (High Acknowledge Flag Off) THEN
                        Reset High Relay
                        High Relay Reset By Button = FALSE
                END IF
        END IF-ELSE
ELSE IF (Top-Level State == Normal Operation Mode ) THEN
        Reset Low Relay
                Reset High Relay
                Low Relay Reset By Button = FALSE
                High Relay Reset By Button = FALSE
END IF-ELSE
```

Figure 4-12: Pseudocode for the Alarm Silence Button ISR.

4.9 DESIGN OF NORMAL OPERATION MODE

4.9.1 Design of Normal Operation Mode for the Sensor Unit

The design of the Sensor Unit software for Normal Operation Mode follows from the requirements and specifications previously developed in Sections 3.1.1 and 3.1.11. Any operation can be initiated in Normal Operation Mode. The Area Monitor will send commands to initiate operations to the Sensor Unit. The Sensor Unit RS-232 ISR will receive commands from the Area Monitor, causing the transition the state transition to perform the requested operation.

The Sensor Unit will compute the PPM when requested to do so by the Area Monitor. The Sensor Unit will compensate for temperature and altitude when calculating the PPM. The Sensor Unit will determine the alarm status after calculating the PPM. The Sensor Unit will send to the Area Monitor the alarm status followed by the PPM.

Figure 4-13, shows the flowchart of the software for Normal Operation Mode. The Sensor Unit RS-232 ISR receives the PPM request from the Area Monitor and sets the send PPM flag. The Normal Operation Mode software calculates the PPM when the send PPM flag is set. The Sensor Unit, to calculate the PPM, first samples the ADC obtaining ADC_VAL. The software triggers the ADC conversion, waits the maximum conversion time of 5us, and then reads the conversion result. Next, the temperature sensor is sampled to obtain the current temperature.

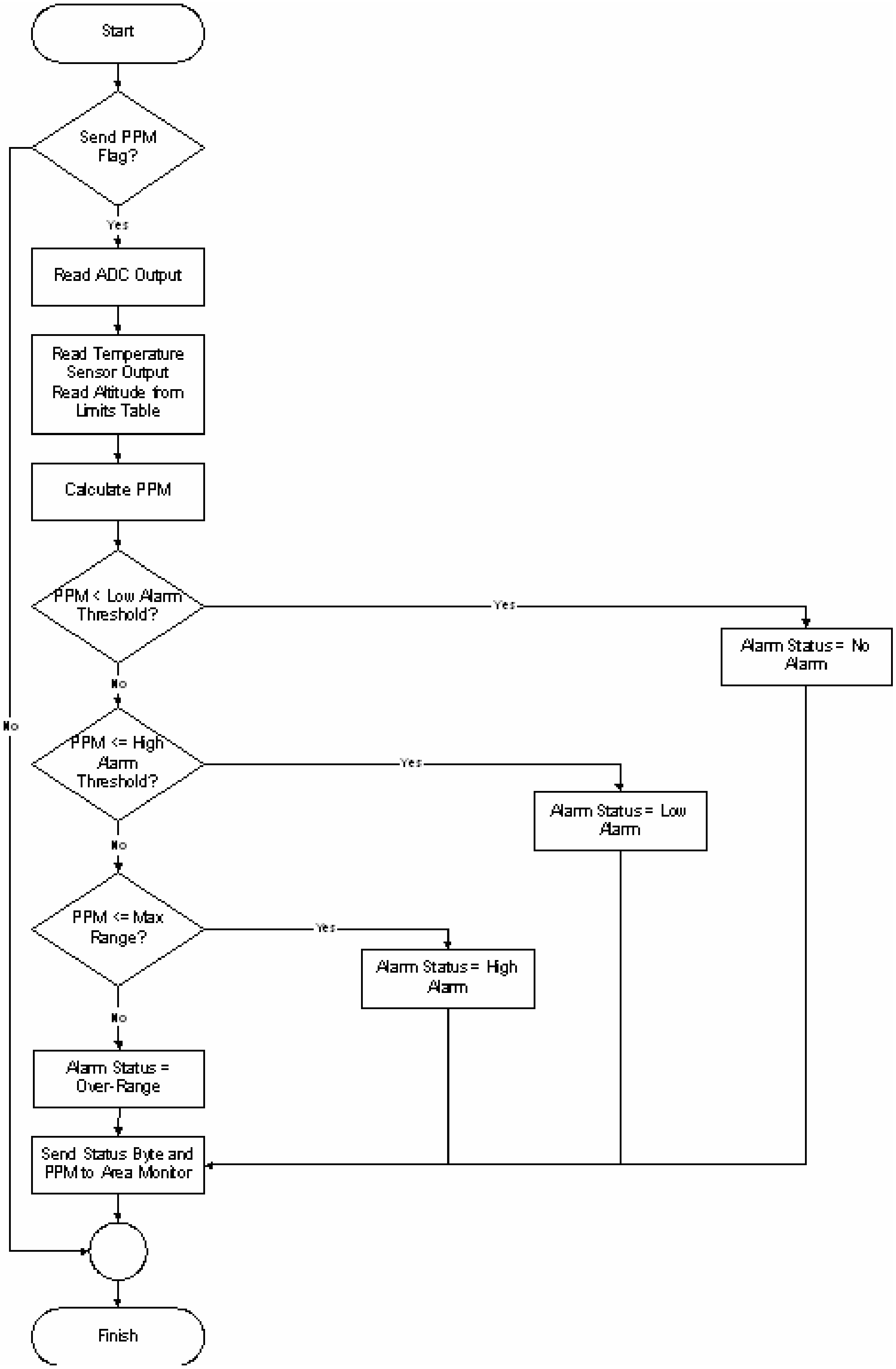

Figure 4-13: Flow chart showing Normal Operation Mode in the Sensor Unit.

The temperature sensor interfaces with the processor using the SPI bus. The processor selects the temperature sensor, asserting the chip select and reads the 16-bit temperature using the SPI protocol. The temperature is represented as an 11-bit sign magnitude integer denoting the temperature in increments of 0.25 C. The least significant 5-bits of the 16-bit reading are not part of the temperature and must be removed. The temperature in degrees Celsius is obtained by diving the 16-bit reading by 32 to remove the least significant 5-bits. Division maintains the sign of the reading, shifting the result right five places would not maintain the sign of the temperature. The resulting value is multiplied by 0.25 to obtain the temperature in degrees Celsius. The altitude is then read from the Limits Table, and the compensated PPM is then computed as described in Section 3.1.1.

The Sensor Unit must then determine the alarm status to send back to the Area Monitor. The alarm status is determined based on the calculated PPM, the low and high alarm thresholds, and the maximum range of the sensor. The alarm status is set to no alarm if the PPM is less than or equal to the low alarm limit. If the PPM is above the low alarm threshold, but less than or equal to the high alarm threshold, the alarm status is set to low alarm. The alarm status is set to high alarm if the PPM is above the high alarm threshold, but not greater than the maximum range of the sensor. The alarm status is set to over-range if the PPM is above the maximum sensor range.

4.9.2 Design of Normal Operation Mode for the Area Monitor

Normal Operation Mode must initiate the PPM calculation every second and must allow the user to initiate any operation. The Area Monitor RS-232 interrupt will receive the alarm status and PPM information from the Sensor Unit, and the keypad interrupt will interpret user input. The remote computer ISR will receive and handle requests from the remote computer. The Area Monitor will use a timer interrupt to track the one-second intervals.

Figure 4-14 shows the top-level software flow for Normal Operation Mode. Normal Operation Mode will first check if it is time to request the next PPM sample, if so the Area Monitor will request the PPM. The timer interrupt will set the Time for PPM flag informing the software that it is time for the next sample. The Area Monitor sends the PPM request to the Sensor Unit and sets the receive buffer to receive the alarm status and PPM when the Time for PPM flag is set.

If it is not time to request the PPM the software will check if the alarm status and PPM have been received. Figure 4-15 shows the flowchart for the code checking for and updating the alarm status and PPM. The incoming Sensor Unit RS-232 ISR will collect the alarm status and PPM, sent from the Sensor Unit, setting the PPM Received flag after receiving the last byte. After the alarm status and PPM are received, the PPM, alarms, and relays must be updated.

The alarm state, the alarm status, and the alarm silence button control the alarms. The alarms will not be active if the alarm status is no alarm. The low alarms will be activated when the alarm status is low alarm. The low and high alarms will be activated if the alarm status is high alarm or over-range.

The two external relays controlled by the system are activated based on the alarm status, and are deactivated based on the alarm state, the alarm silence button, and the latching and acknowledge settings. The alarm silence button provides the user with the means to turn off the alarm horns, and depending on the acknowledge values the relays can be disabled as well.

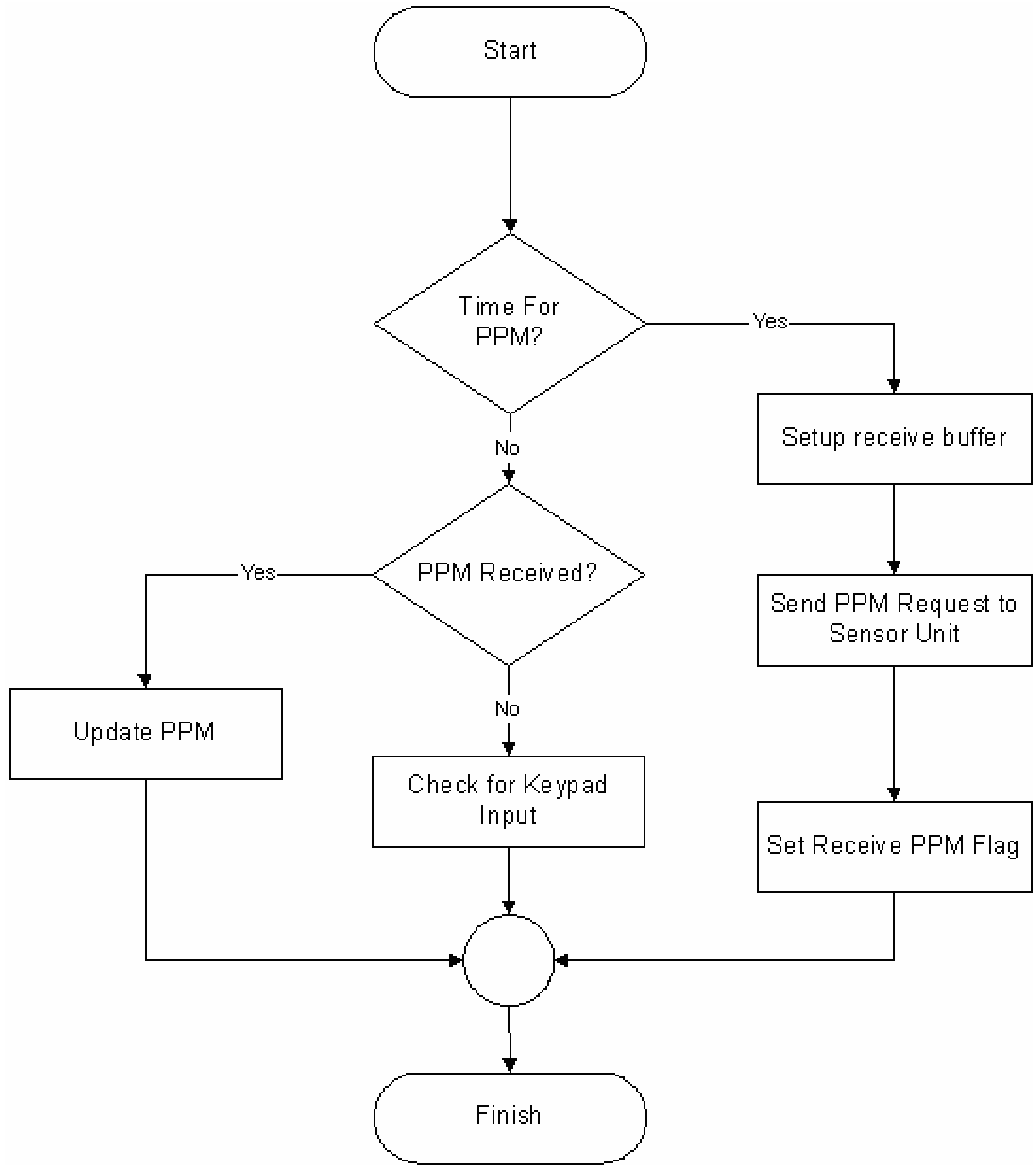

Figure 4-14: Flowchart of the top-level code for Normal Operation Mode for the Area Monitor.

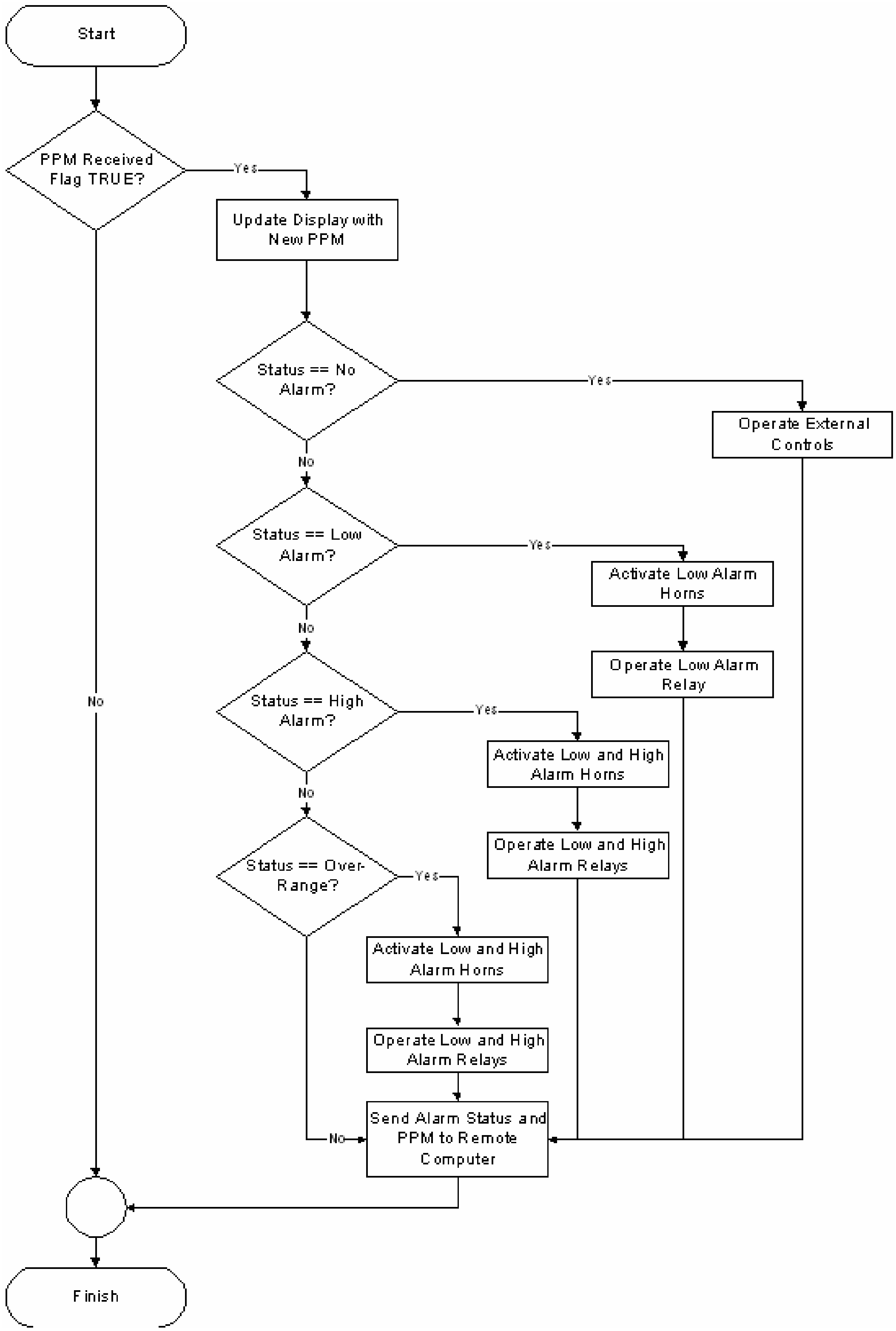

Figure 4-15: Flowchart of the code to handle the newly received status and PPM.

The low relay is activated when the alarm status is low alarm and the low and high relays are activated when the alarm status is high alarm or over-range. The high relay is deactivated when leaving the high alarm state, entering low alarm state, if the high relay latching is not set, if the high relay latching is set the relay will remain active. Similarly, the low relay is deactivated when entering no alarm state if the low relay latching is not set, otherwise the low relay will remain activated.

The built-in user interface menu system must also be maintained in Normal Operation Mode. The Normal Operation Mode software will maintain and update the menu system based on the user input from the keypad. If the alarm status and PPM has not been received, Normal Operation Mode checks for and handles keypad input. The keypad controls the menu system of the built-in interface. The keypad ISR after detecting a key press sets flags indicating that a key press occurred and the key that was pressed. Figure 4-16 shows the flowchart of the code to detect and handle keypad input. The menus are traversed as described in Section 3.1.11. The enter key switches the menu state from PPM Display to Menu Mode and initiates the currently selected operation when in Menu Mode. The up and down keys will cycle forwards and backwards respectively, through the menus as shown in Figure 3-18. The left and right keys will switch the menu state from Menu Mode back to PPM Display.

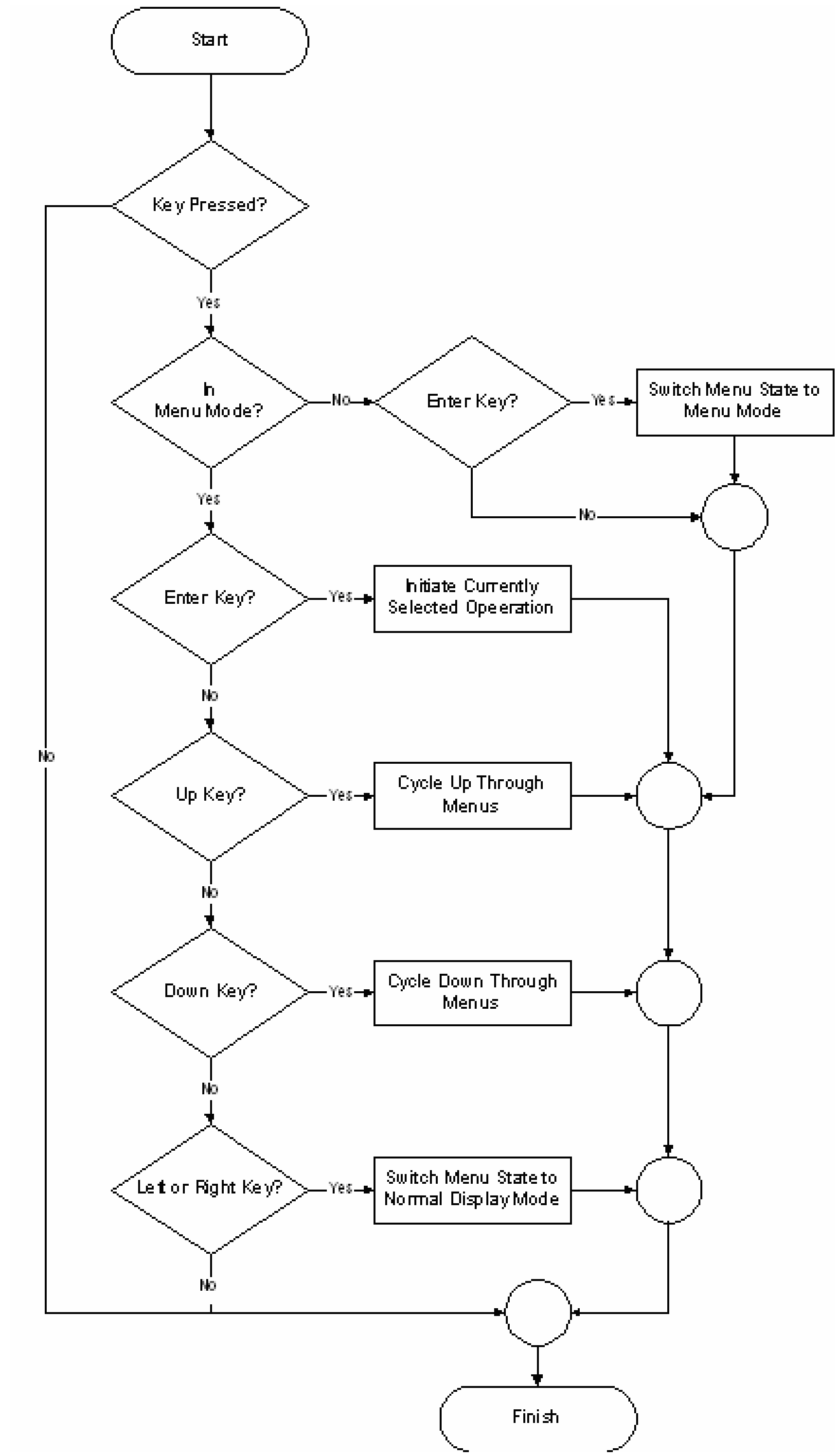

Figure 4-16: Flowchart of the code to check for and handle a keypad input.

4.10 DESIGN OF ALARM MODE

4.10.1 Design of Alarm Mode for the Sensor Unit

Alarm Mode is identical to Normal Operation Mode except that the alarms are activated and the user can only initiate a Configuration. Alarm Mode will calculate the PPM or perform a Configuration in response to a request from the Area Monitor. The Sensor Unit RS-232 ISR will handle the requests from the Area Monitor. If the configuration command is received the Sensor Unit RS-232 ISR will cause the transition to Configuration Mode and set up the buffer to receive the configuration data. If the request PPM message is received the Sensor Unit RS-232 ISR will set the Send PPM flag, informing the Alarm Mode software that the PPM sample must be taken. The calculation and sending of the alarm status and PPM is identical to the process described in the Section 4.9.1.

4.10.2 Design of Alarm Mode for the Area Monitor

Alarm Mode is similar to Normal Operation Mode only the alarms are activated and the operations that the user can initiate are limited to a Configuration. The Alarm Mode routine will check if it is time to request the next PPM and if so will request it. The request procedure is identical to the Normal Operation Mode code, shown above in Figure 4-14. As in Normal Operation Mode, the Area Monitor ISR will collect the alarm status and PPM, setting the Time for PPM flag after receiving the last byte. If the Time for PPM flag is not set, Alarm Mode will check if the alarm status and PPM has been received from the Sensor Unit. The alarm status and PPM is handled identically as in Normal Operation Mode, as shown in Figure 4-15 above.

If the PPM is not requested or received, the software will check if keypad input has been detected. Alarm Mode will only allow the user to initiate a Configuration while in Alarm Mode. The keypad ISR will handle keypad input and will set the appropriate flags in when a key press is detected. Keypad input will be handled based on the Alarm Mode menu system developed in Section 3.1.11. Figure 4-17 shows the flowchart for updating the Alarm Mode menus. The enter key will switch from PPM Display to Menu Mode if currently in PPM Display, and will initiate the configuration operation if currently in Menu Mode. The up and down keys have no effect on the menus because only one option can be selected in Alarm Mode. The left and right keys will switch from Menu Mode back to Display PPM.

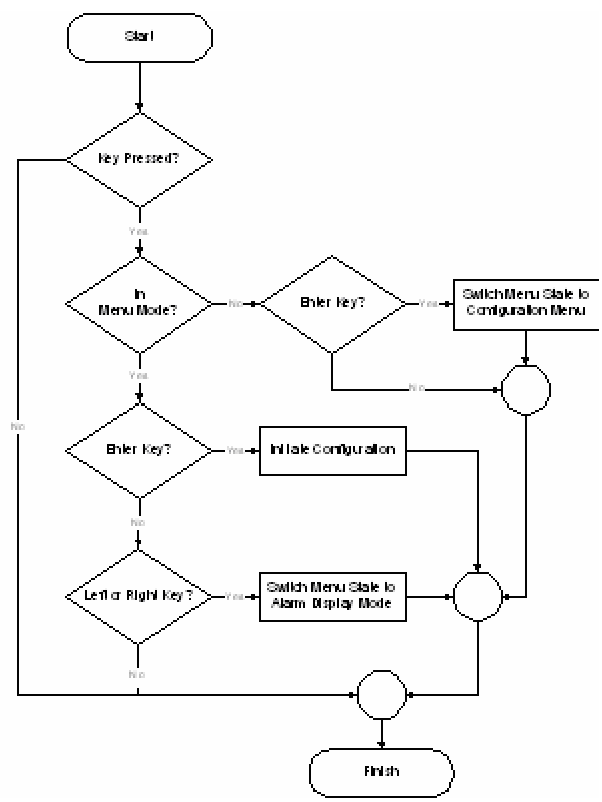

Figure 4-17: Flowchart of the code to update the Alarm Mode menus.

4.11 DESIGN OF DIAGNOSTIC ROUTINE

The diagnostic will test the functionality of critical components in the system. The diagnostic routine will perform the tasks discussed in the requirements and specifications developed in Section 3.1.3. The Sensor Unit will perform the diagnostic tests and will report the results to the Area Monitor. The Area Monitor will report the results to the user.

4.11.1 Design of Diagnostic Routine for the Sensor Unit

Diagnostic Mode is modeled as a finite state machine with three states. This first state, Sensor Diagnostic, performs the two diagnostics on the Sensor Module and sends the diagnostic test command to the Area Monitor. The next state, Communication Test, waits for the diagnostic test command response from the Area Monitor completing the diagnostic handshake. The final state, Send Report, compiles and sends the diagnostic report to the Area Monitor.

The Sensor Unit RS-232 ISR will receive the diagnostic command from the Area Monitor causing the transition into Diagnostic Mode, setting the diagnostic state to Sensor Diagnostic. The diagnostic routine will check if the Sensor Module has expired, if the data in the Sensor Module FLASH has been corrupted, and if the communications link between the Area Monitor and Sensor Unit is operational. The diagnostic report is then compiled and sent to the Area Monitor. Figure 4-18 shows the flowchart for the diagnostic routine code.

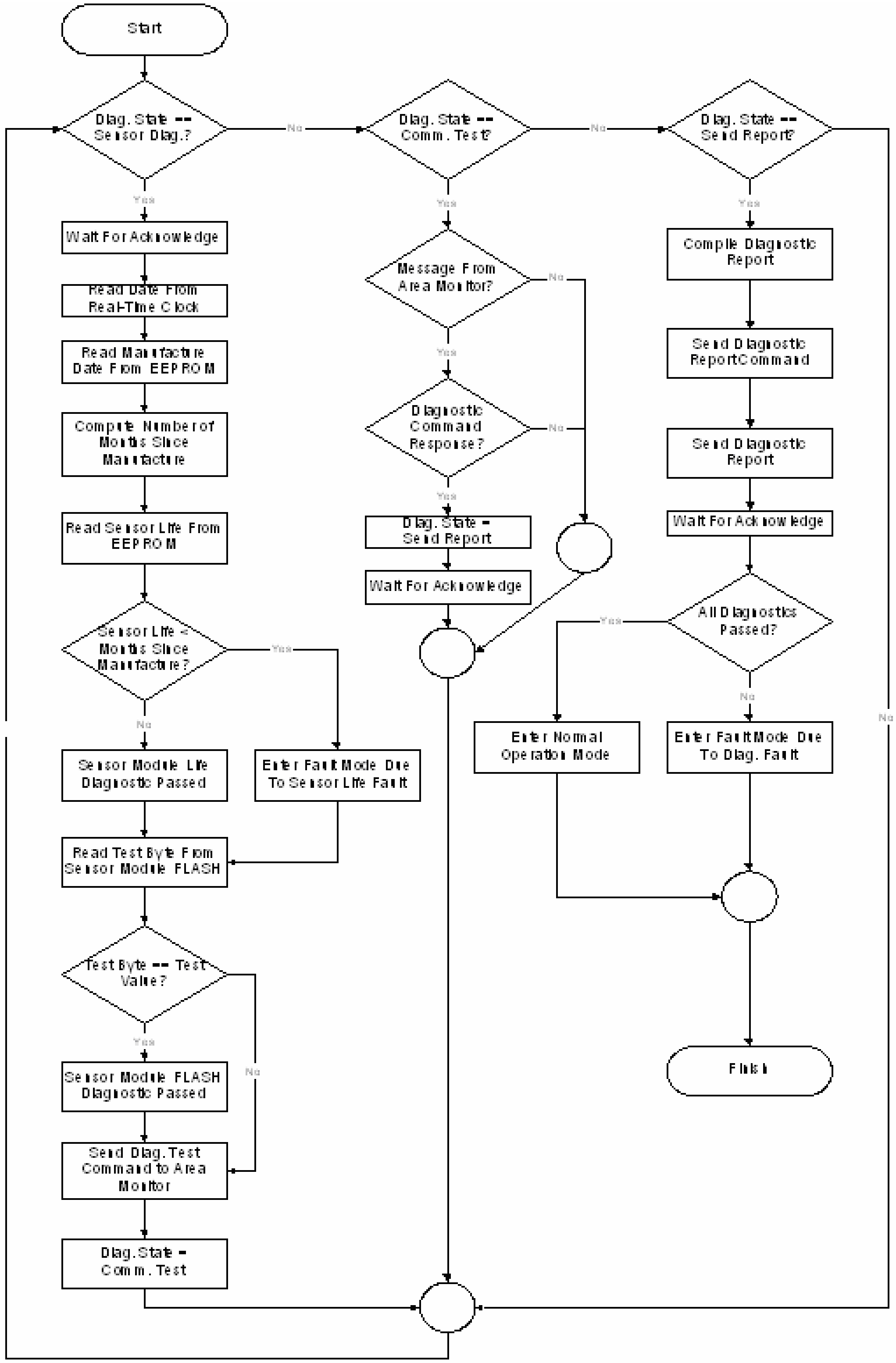

Figure 4-18: Flowchart of the Sensor Unit diagnostic routine.

The Sensor Unit will wait for the acknowledge message to be received from the Area Monitor before performing the Sensor Diagnostics. The Sensor Module diagnostics are performed first. The diagnostic routine will first check to verify that the Sensor Module has not expired. This diagnostic passes if the Sensor Module has not expired, and fails if the Sensor Module has expired. The Sensor Module has expired if the number of months since manufacture is greater than the sensor life. To determine if the Sensor Module has expired, the current month and year are read from the real-time clock.

The real-time clock uses an Intel style communication protocol. The processor will trigger the read of the month and year by providing the address of the register containing the month and year. The processor then waits the required 100ns before reading the data from the real-time clock.

Next, the Sensor Unit reads the Sensor Module manufacture month and year, and the sensor life from the Sensor Data Table in the EEPROM. The number of months since the manufacture of the Sensor Module is computed by subtracting the manufacture date from the current date. Equation 8 shows the calculation of the number of months since manufacture of the Sensor Module.

$$Months\ Since\ Manufacture = \left[12 \times \left(Year_{Current} - Year_{Manufacture}\right)\right] + \left(Month_{Current} - Month_{Manufacture}\right)$$

Equation 8

The diagnostic routine then verifies the integrity of the Sensor Module FLASH. This diagnostic passes if the test value read from the Sensor Module FLASH equals the known test value, otherwise this diagnostic fails. The communications link between the Area Monitor and the Sensor Unit is tested after the Sensor Module diagnostics have completed. After completing this test, the diagnostic test command is sent to the Area Monitor, and the diagnostic state is updated to Communications Test, to perform the diagnostic on the communication link.

The Sensor Unit must receive the diagnostic test command response within 60 seconds of sending the diagnostic test command or the communications diagnostic fails. The Area Monitor will respond with the diagnostic test command response. The Sensor Unit RS-232 ISR will set the Message Received Flag and store the byte received in the Message global variable. If the byte received in the diagnostic command response, the diagnostic state is updated to Send Report, to send the diagnostic report to the Area Monitor.

The diagnostic report must be sent to the Area Monitor. The Sensor Unit first sends the diagnostic report command to the Area Monitor. The Sensor Unit waits to receive the acknowledge message from Area Monitor after sending the Diagnostic Report Command. The diagnostic report command informs the Area Monitor that the next byte received is the diagnostic report. Sending the diagnostic report command first is required to prevent the Area Monitor from interpreting the diagnostic report as another message. The Sensor Unit compiles the diagnostic report and waits for the Area Monitor to reply with the acknowledge message, indicating receipt of the diagnostic command. The diagnostic report will contain a pass/fail report for each of the three diagnostics. The diagnostic report is then sent to the Area Monitor. The Sensor Unit enters Fault Mode, due to a diagnostic fault if any of the three diagnostic does not pass. If all three diagnostics pass, the Sensor Unit enters Normal Operation Mode, after receiving the acknowledge command from the Area Monitor.

4.11.2 Design of Diagnostic Routine for the Area Monitor

The software to control the Area Monitor while performing a diagnostic is designed based on the specifications and requirements described in Section 3.1.3. The Area Monitor initiates the diagnostic by sending the diagnostic command to the Sensor Unit. The Area Monitor will not perform any of the diagnostic tests, but will participate in the communications diagnostic.

The diagnostic routine is modeled as a finite state machine with three states. The first state, Communication Test, waits to receive the diagnostic test command from the Sensor Unit. The second state, Report Command, waits for the Sensor Unit to send the Diagnostic Report Command. The final state, Diagnostic Report, waits to receive the diagnostic report.

Figure 4-19 shows the flowchart for the diagnostic mode on the Area Monitor. After sending the diagnostic command to the Sensor Unit initiating the diagnostic, the Area Monitor will wait for the diagnostic test command, sending the acknowledge message when entering the Communication Test state. The Sensor Unit initiates the diagnostic communication test by sending the diagnostic test command to the Area Monitor. The Area Monitor RS-232 ISR will retrieve and inform the main software when a message from the Sensor Unit has arrived. The Area Monitor will check to see if the message received is the diagnostic test command. If the message is the diagnostic test command, the Area Monitor will respond, sending the diagnostic test command response to the Sensor Unit completing the diagnostic handshake testing the communication link between the Area Monitor and the Sensor Unit. The Area Monitor updates the diagnostic state to Report Command.

The Area Monitor waits for the diagnostic report command to be received from the Sensor Unit. The diagnostic report command is required to prevent the Area Monitor from mistaking the diagnostic report data for another message. The diagnostic report command informs the Area Monitor that the next byte received is the diagnostic report. Mistaking the diagnostic report data for another message could cause the Area Monitor to crash. When the Area Monitor receives the diagnostic report command, the diagnostic report command is sent to the remote computer, and the acknowledge message is sent to the Sensor Unit. The diagnostic state is updated to Diagnostic Report to receive the diagnostic report.

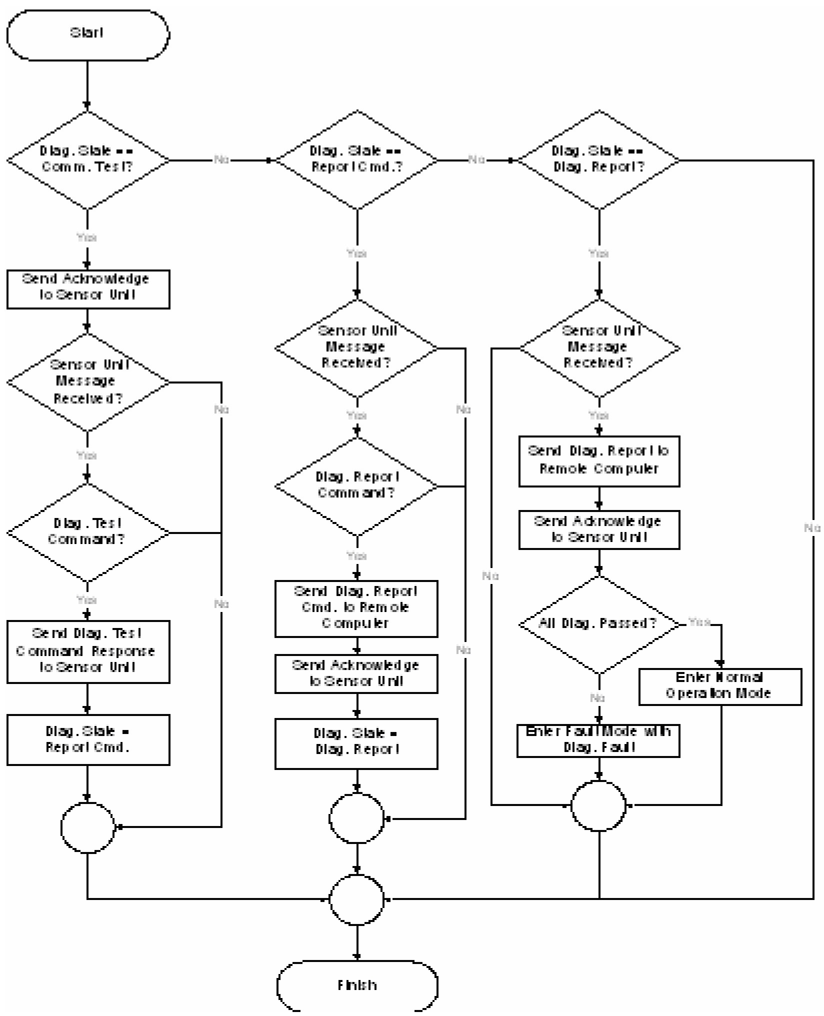

Figure 4-19: Flowchart showing the code for the Area Monitor diagnostic routine.

The Area Monitor waits to receive the diagnostic report data. The diagnostic report data is the next byte sent by the Sensor Unit after the diagnostic report command. After receiving the diagnostic report data, the Area Monitor sends the diagnostic report data to the remote computer, and sends the acknowledge message to the Sensor Unit. The Area Monitor checks the diagnostic report data to verify that all diagnostics passed. If any diagnostic failed, the Area Monitor enters Fault Mode due to a diagnostic fault. If all diagnostics passed, the Area Monitor enters Normal Operation Mode, after sending the acknowledge message to the Sensor Unit.

4.12 DESIGN OF POWER-UP MODE

4.12.1 Design of Power-Up Mode for the Sensor Unit

The Sensor Unit will enter Power-Up Mode when power is first applied to the system. Power-Up Mode will perform two initial checks on the system before entering Warm-Up Mode. The writing EEPROM flag is tested to determine if power was lost while a write to EEPROM was taking place and the presence of the Sensor Module is verified.

Figure 4-20 shows the flowchart for the Power-Up Mode. Power-Up Mode first checks if the Sensor Module is present by reading the value from an input pin. The value is a ‘0’ when the Sensor Module is present and a ‘1’ when it is absent. If the Sensor Module is not present, the Sensor Unit enters Fault Mode and a Sensor Module Missing Fault is issued. If the Sensor Module is present, a Loss of Power Fault is checked for.

The Writing EEPROM flag is read from the onboard EEPROM. If the Writing EEPROM flag is set, then the Sensor Unit was writing to EEPROM when power was lost and the Sensor

Unit enters Fault Mode, issuing a Loss of Power Fault. If the Writing EEPROM flag is not set, then the no Loss of Power Fault occurred and the system enters Warm-Up Mode, sending the warm-up message to the Area Monitor.

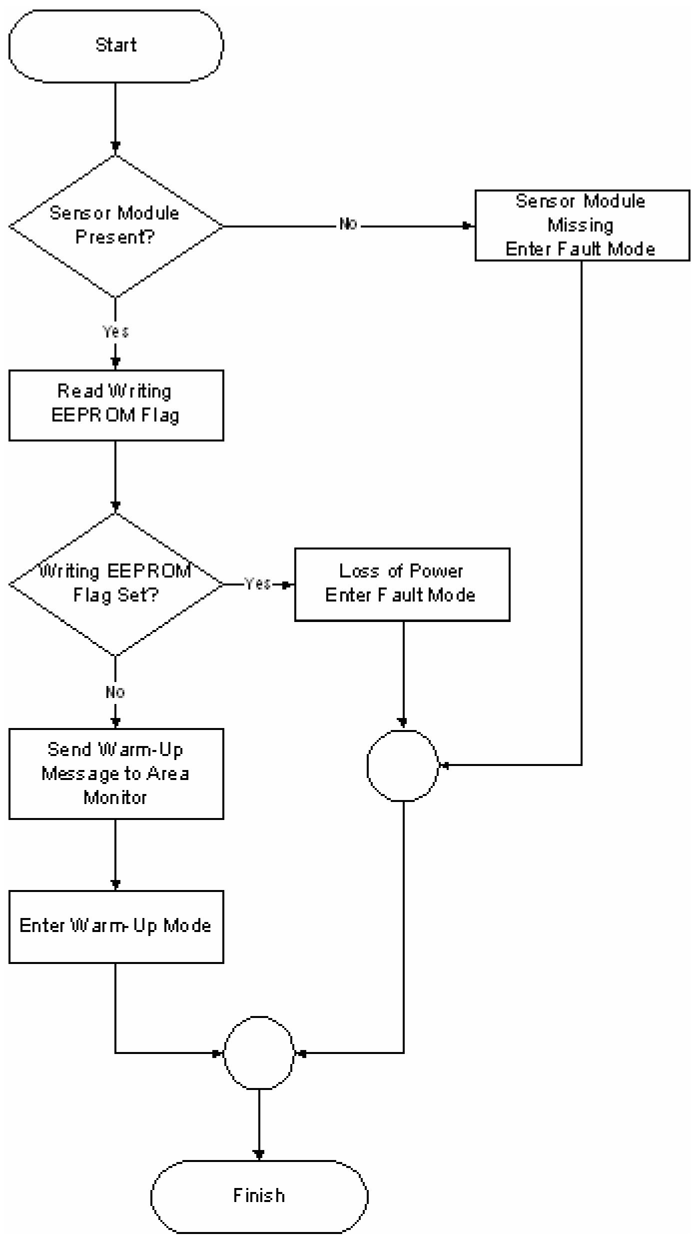

Figure 4-20: Flowchart showing the code for Power-Up Mode for the Sensor Unit.

4.12.2 Design of Power-Up Mode for the Area Monitor

The Area Monitor will enter Power-Up Mode when power is first applied to the system. The power-up message is displayed on the 4-character display, and the power-up message will be sent to the remote computer to informing the user that the Area Monitor is in Power-Up Mode. The Area Monitor will remain in Power-Up Mode until receiving the warm-up message or a fault message. The Area Monitor RS-232 ISR will handle the state change when a fault message arrives. The Area Monitor enters Fault Mode, in response to either a Sensor Module Missing Fault Message or a Loss of Power Fault Message, and enters Warm-Up Mode in response to the warm-up message.

4.13 DESIGN OF SENSOR WARM-UP MODE

4.13.1 Design of Sensor Warm-Up Mode for the Sensor Unit

As previous discussed in Section 3.1.7, Warm-Up Mode will determine what data is required from the Sensor Module, will check to verify that the Sensor Module is not expired, and will evaluate the Magnitude and Slope tests to determine when the Sensor Module is ready for operation. After passing both of the Warm-Up Tests, the system enters Diagnostic Mode to ensure full functionality.

Figure 4-21 shows the pseudocode for Warm-Up Mode. Upon entering Sensor Warm-Up Mode, the Sensor Unit waits for the acknowledge message from the Area Monitor. After receiving the acknowledge message from the Area Monitor, the Sensor Unit checks to ensure that the Sensor Module has not expired.

```
Wait for Acknowledge

Download Manufacture Month From Sensor Module FLASH
Download Manufacture Year From Sensor Module FLASH
Download Serial Number From Sensor Module FLASH
Download Sensor Type From Sensor Module FLASH
Download Shelf Life From Sensor Module FLASH
Download Service Life From Sensor Module FLASH

Current Month = Read Current Month from Real-Time Clock
Current Year = Read Current Year from Real-Time Clock
SensorLife = Sensor service life in months

IF( [(Shelf Life + Service Life) - (Current Date - Manufacture Date)] > 0 ) THEN

    IF( Same Sensor Serial Number and Sensor Type ) THEN
        No Data Must be Downloaded from Sensor Module FLASH
    ELSE IF( Different Serial Number AND Same Type ) THEN
        Recalculate Sensor Life Remaining and Use Other Existing Data
        IF( [Current Date - Manufacture Date] < Shelf Life ) THEN
                SensorLife = Service Life
        ELSE
                SensorLife = Shelf Life + Service Life - [Current Date - Manufacture Date]
        END IF
    ELSE IF( Different Serial Number AND Different Sensor Type ) THEN
        Download All Data and Calculate Sensor Life Remaining
        IF( [Current Date - Manufacture Date] < Shelf Life ) THEN
                SensorLife = Service Life
        ELSE
                SensorLife = Shelf Life + Service Life - [Current Date - Manufacture Date]
        END IF
    END IF

    Write Configuration, Calibration, and Sensor Data Tables To EEPROM
    Send Configuration, Calibration, and Sensor Data Tables To the Area Monitor

    Collect Initial 9 Readings From the ADC
    Tests Passed = FALSE

    WHILE( Tests Passed == FALSE )
        IF( Time For Next Sample ) THEN
                CurrentSample = ADC Reading
                IF( CurrentSample < Magnitude Test Threshold ) THEN
                        Slope = Slope Found From Least Squares Fit to the 10 Readings

                        IF( Slope < Slope Test Threshold ) THEN
                                Tests Passed = TRUE
                        END IF
                END IF
        END IF

    END WHILE
    Send Diagnostic Command to Area Monitor
    Enter Diagnostic Mode

ELSE
    Enter Fault Mode Due to Sensor Module Past Lifetime Fault
END IF
```

Figure 4-21: Pseudocode for Warm-Up Mode in the Sensor Unit.

The Sensor Unit month and year of the Sensor Module manufacture date along with the Sensor Module Serial Number and Sensor Type are read from the Sensor Module FLASH. The current month and year are read from the real-time clock. These values are used to check if the Sensor Module has expired and to determine what data must be downloaded.

The Sensor Module FLASH communicates with the processor using the SPI bus. The processor triggers the read by asserting the Sensor Module FLASH chip select and sending the read command to the FLASH chip. The processor then sends 24-bit address the address (most significant byte first) of the desired byte of data. The processor then performs a read of the SPI bus to acquire the data from the FLASH chip.

The first test will check if the Sensor Module has expired. The Sensor Module has expired if the number of months since manufacture is greater than the shelf life plus the service life of the Sensor Module. The number of months since manufacture of the Sensor Module is the difference between date of manufacture of the Sensor Module and the current date. If the Sensor module has expired the Sensor Unit enters Fault Mode and a Sensor Past Lifetime Fault is issued.

The Sensor Serial Number and Sensor Type stored in the Sensor Unit EEPROM and the Sensor Module Serial Number and Type are compared to determine what data, if any, must be downloaded from the Sensor Module FLASH. The data downloaded based on the serial numbers and sensor types is defined in Section 3.1.7. All floating-point values are converted from the IEEE 754 floating-point format to the Microchip floating-point format. After downloading and writing all required data to EEPROM, the data in the working Configuration, working Calibration, and Sensor Data Tables is sent to the Area Monitor.

The software will then monitor the sensor output to determine when the Sensor Module passes the Magnitude Test and Slope Tests and is ready for operation. The Magnitude Test checks if the sensor output is below the manufacturer's threshold. The Slope Test is performed only if the current reading first passes the Magnitude Test. The Slope Test checks if the slope of the first order polynomial least squares fit to the 10 samples is less than the manufacturer's threshold. A 10-sample window with one sample occurring every second is used. Ten samples are required for the Slope Test, so the first 9 samples are collected without performing either test. Then for every new sample, the Magnitude and Slope tests are evaluated. After both tests pass, the Test Passed flag is set and the testing loop exits. The diagnostic start command is sent to the Area Monitor and the Sensor Unit enters Diagnostic Mode.

4.13.2 Design of Sensor Warm -Up Mode for the Area Monitor

The Area Monitor will inform the user that the system is currently in Warm-Up Mode, displaying the warm-up message on the 4-character display. The software for Warm-Up Mode is modeled as a finite state machine, with two states. The first state, Receive New Sensor Data, will receive the data sent from the Sensor Unit. The second state, Wait For Diagnostic, will wait for the diagnostic start command to be received from the Sensor Unit.

Figure 4-22 shows the pseudocode for Warm-Up Mode. The Area Monitor waits until all new data from the Sensor Module is received, sending the acknowledge message to the Sensor Unit when entering the Receive New Sensor Data state. The Area Monitor RS-232 ISR will collect the data setting the All Data Received Flag after receiving the last byte of data. The number of bytes of data sent is the size in bytes of the Configuration, Calibration, and Sensor

Data Tables. When the All Data Received Flag is set, the Warm-Up Mode software writes the data to the Area Monitor EEPROM and updates the Warm-Up State to Wait for Diagnostic.

In the Wait for Diagnostic state, the Area Monitor will wait for the diagnostic start command. The Area Monitor RS-232 ISR will set the Message Received Flag and will store the last byte received in the Message variable. After receiving the diagnostic start command, the diagnostic start command is sent to the remote computer and the Area Monitor enters Diagnostic Mode.

```
Display Warm-Up Message
IF( Warm-Up State == Receive New Sensor Data ) THEN
        Send Acknowledge to Sensor Unit
        IF( All Data Received Flag Set ) THEN
                Write Data to Configuration, Calibration, and Sensor Data Tables
                Warm-Up State = Wait for Diagnostic
        END IF
END IF
ELSE IF( Warm-Up State == Wait for Diagnostic ) THEN
        IF( Sensor Unit Message Received ) THEN
                IF( Message == Diagnostic Start Command ) THEN
                        Send Diagnostic Start Command to Remote Computer
                        Enter Diagnostic Mode
                END IF
        END IF
END IF
```

Figure 4-22: Psuedocode of the Area Monitor Warm-Up Mode.

4.14 DESIGN OF CONFIGURATION MODE

4.14.1 Design of Sensor Unit Configuration Mode

Configuration Mode, discussed in Section 3.1.4, allows the user to update operational settings of the system. The Area Monitor triggers the entry into Configuration Mode in the Sensor Unit by sending the configuration command. The Sensor Unit waits until all configuration data has been received from the Area Monitor. The Sensor Unit RS-232 ISR will receive the configuration data, setting the All Configuration Data Received flag after receiving the last byte of configuration data from the Area Monitor before updating the configuration tables. The number of bytes of configuration data sent is equal to the number of bytes in the Configuration Table.

After all configuration data has been received the Sensor Unit backs up the previous 1 configuration and the working configuration tables, and updates the working configuration table with the new configuration table. The previous 1 configuration and the working configuration tables, must be stored to maintain the required historical data. The previous 2 configuration table is overwritten with the previous 1 configuration table, and the working configuration table overwrites the previous 1 configuration table. The new configuration data is written into the working configuration table. After updating the three configuration tables, the Sensor Unit sends the configuration complete ok message to the Area Monitor. The Sensor Unit then waits for the Area Monitor to send the acknowledge message, acknowledging receipt of the configuration complete ok message. After receiving the acknowledge message from the Area Monitor, the Sensor Unit enters Normal Operation Mode.

```
Send Acknowledge Message to Area Monitor
IF( All Configuration Data Received ) THEN

    Copy Previous 1 Configuration Table Into Previous 2 Configuration Table
    Copy Working Configuration Table Into Previous 1 Configuration Table

    Write New Configuration Data to Working Configuration Table

    Send Configuration Complete OK to Area Monitor

    Wait For Acknowledge from Area Monitor

    Enter Normal Operation Mode

END IF
```

Figure 4-23: Pseudocode for Sensor Unit Configuration Mode

4.14.2 Design of Area Monitor Configuration Mode

The Area Monitor contains two separate functions to perform a configuration. One function will perform the configuration when the built-in interface is used and the second function will perform the configuration when the remote computer is used. The Area Monitor will collect the configuration data from the user and then send the data to the Sensor Unit. The design of the function to perform the configuration using the built-in interface is presented first, followed by the function to handle a configuration from the remote computer.

The software for Configuration Mode using the built-in interface is modeled as a finite state machine with ten states. Figure 4-24 shows the pseudocode for Configuration Mode when using the built-in man-machine interface.

```
Config_State = Current State within Configuration Routine (Global Variable)
Data = Data collected from user, updated by the keypad ISR (Global Variable)
KeyPress = Keypres detected by Keypad ISR from the Keypad (Global Variable)

IF( Config_State == Prompt Low Alarm ) THEN
     Display Low Alarm Prompt
     IF( KeyPress == ENTER ) THEN
               Config_State = Enter Low Alarm
               Data = Read Low Alarm Threshold From EEPROM
     END IF
     KeyPress = NULL
ELSE IF( Config_State == Enter Low Alarm ) THEN
     Display Data To User
     IF(KeyPress == ENTER ) THEN
               IF( Data <= Maximum Sensor Range ) THEN
                         Store Low Alarm Threshold Data in Buffer
                         Config_State = Prompt High Alarm
               END IF
               KeyPress = NULL
     END IF
ELSE IF( Config_State == Prompt High Alarm ) THEN
     Display High Alarm Prompt
     IF( KeyPress == ENTER ) THEN
               Config_State = Enter High Alarm
               Data = Read High Alarm Threshold from EEPROM
     END IF
     KeyPress = NULL
ELSEIF( Config_State == Enter High Alarm ) THEN
     Display Data To User
     IF(KeyPress == ENTER ) THEN
               IF( (Data > Low Alarm Threshold) AND (Data <= Maximum Sensor Range) ) THEN
                         Store High Alarm Threshold Data in Buffer
                         Config_State = Prompt Zero Calibration Duration

               END IF
               KeyPress = NULL
     END IF
ELSE IF( Config_State == Prompt Zero Calibration Duration ) THEN
     Display Zero Calibration Duration Prompt
     IF( KeyPress == ENTER ) THEN
               Config_State = Enter Zero Calibration Duration
               Data = Read Zero Calibration Duration from EEPROM
     END IF
     KeyPress = NULL
ELSE IF( Config_State == Enter Zero Calibration Duration ) THEN
     Display Data To User
     IF(KeyPress == ENTER ) THEN
               IF((Data >= 2) AND (Data <= 5) ) THEN
                         Store Zero Calibration Duration Data in Buffer
                         Config_State = Prompt Reference Calibration Duration
               END IF
               KeyPress = NULL
     END IF
ELSE IF( Config_State == Prompt Reference Calibration Duration ) THEN
     Display Reference Calibration Duration Prompt
     IF( KeyPress == ENTER ) THEN
               Config_State = Enter Reference Calibration Duration
               Data = Read Reference Calibration Duration from EEPROM
     END IF
     KeyPress = NULL
ELSE IF( Config_State == Enter Reference Calibration Duration ) THEN
     Display Data To User
     IF(KeyPress == ENTER ) THEN
               IF((Data >= 2) AND (Data <= 10)) THEN
                         Store Reference Calibration Duration Data in Buffer
                         Config_State = Send Data
               END IF
               KeyPress = NULL
     END IF
ELSE IF( Config_State == Send Data ) THEN
     Send Configuration Command to the Sensor Unit
     Wait For Acknowledge from Sensor Unit
     Write Buffer Contents to the Working Configuration Table
     Send Buffer Contents to Sensor Unit
     Config_State = Wait For Acknowledge
ELSE IF( Config_State == Wait For Acknowledge ) THEN
     IF( Sensor Unit Message Received ) THEN
               IF( Message == Configuration Complete OK ) THEN
                         Send Configuration Complete Ok  to Remote Computer
                         Send Acknowledge Message to Sensor Unit
                         Set Top-Level State to Normal Operation Mode
               END IF
     END IF
END IF-ELSE
```

Figure 4-24: Pseudocode of the Area Monitor Configuration routine for the built-in interface.

Configuration Mode will prompt the user to enter each piece of data. The user must press the enter key to change to the state to update the data. When the user presses the enter key, the current setting is read from the working configuration table and displayed to the user on the 4-character display. The user will use the up, down, left, and right keys to alter the data. The keypad ISR will detect key presses, update the data, and set the KeyPress flag indicating which key was pressed. Verification of each data element against the limits described in Section 3.1.4 occurs after the user presses the enter key accepting the value. The configuration routine will not move on to the next data entry until the currently entered data value is determined to be within the specified limits. If the data element is within the limits that data is stored in the temporary buffer and the configuration state is updated to next the state prompting the user for the next setting. After the reference calibration duration is entered and verified, the configuration state is updated to send data.

After all data has been entered, the Area Monitor will update the working configuration table in EEPROM to reflect the new values. After updating the table, the Area Monitor sends the new configuration data to the Sensor Unit. The configuration command is sent to the Sensor Unit and the Area Monitor waits for the Sensor Unit to acknowledge that command. After receiving the acknowledge, the Area Monitor sends the configuration data to the Sensor Unit. The low alarm threshold is sent first, followed by the high alarm threshold, zero calibration duration, and finally the reference calibration duration. The Configuration State is updated to Wait for Acknowledge.

The Area Monitor then waits to receive the configuration complete ok message from the Sensor Unit. When the configuration complete message is received, the Area Monitor sends the configuration complete message to the remote computer, sends the acknowledge message to the Sensor Unit, and then enters Normal Operation Mode.

Configuration Mode can also be performed from the remote computer. Figure 4-25 shows the pseudocode for the Remote Computer Configuration Mode. Configuration Mode will display the configuration message indicating the remote computer is performing a configuration. The Configuration Mode collects the configuration data from the remote computer, converts the low and high alarm thresholds from IEEE 754 floating-point format to Microchip floating-point format, and sends the configuration data to the Sensor Unit. The remote computer RS-232 ISR will collect the configuration data, setting the All Configuration Data Received flag after the last byte of data has been received. The conversion from IEEE 754 to Microchip floating-point format of the low and the high alarm thresholds occurs after receiving all the configuration data. The Area Monitor then sends the configuration command to the Sensor Unit over the RS-232 link and waits for the Sensor Unit to acknowledge. The Area Monitor then writes the new configuration data to the working configuration table in EEPROM. Next, the Area Monitor sends the new configuration data to the Sensor Unit in the same order as when using the built-in interface.

The Area Monitor then waits to receive the configuration complete ok message from the Sensor Unit. The Area Monitor RS-232 ISR sets the Sensor Unit Message Received Flag and stores the received byte in the Message variable after receiving a byte from the Sensor Unit. After receiving the configuration complete ok message, the Area Monitor sends the configuration complete ok message to the remote computer and then sends the acknowledge message to the Sensor Unit. The Area Monitor then enters Normal Operation Mode.

```
Display Configuration Message
IF( Config_State == Wait For Data ) THEN
     IF( All Configuration Data Received ) THEN
               Convert Low Alarm Threshold to Microchip Floating-Point
               Convert High Alarm Threshold to Microchip Floating-Point

               Send Configuration Command to Sensor Unit

               Write New Configuration Data to Configuration Table

               Send New Configuration Data to the Sensor Unit

               Config_State = Wait For Acknowledge
     END IF
ELSE IF( Config_State == Wait For Acknowledge ) THEN
     IF( Sensor Unit Message Received ) THEN
               IF( Message == Configuration Complete OK ) THEN
                         Send Configuration Complete Ok to Remote Computer
                         Send Acknowledge Message to Sensor Unit
                         Set Top-Level State to Normal Operation Mode
               END IF
     END IF
END IF-ELSE
```

Figure 4-25: Pseudocode for the Area Monitor Configuration routine for the remote computer.

4.15 DESIGN OF ZERO CALIBRATION MODE

4.15.1 Design of Sensor Unit Zero Calibration

As described in Section 3.1.2.2, the Sensor Unit performs the calculations to obtain the new zero point, uses the zero point to calculate an estimated reference point and then the SLOPE. The Area Monitor initiates the Zero Calibration sending the zero calibration command to the Sensor Unit. The system will enter Fault Mode if the zero point and/or the reference point are not observed within the specified zero duration, or if the zero point or SLOPE are outside the manufacturer's limits.

Figure 4-26 shows the pseudocode for Zero Calibration Mode. Upon entering zero calibration, the Sensor Unit will send the acknowledge message to the Area Monitor. The Sensor Unit waits to begin monitoring for the zero point until receiving the zero calibration start command from the Area Monitor. This allows the technician time to connect any required equipment to the system without deducting that time from the zero calibration duration.

After receiving the zero calibration start command the Sensor Unit sends the acknowledge message to the Area Monitor. The zero calibration duration is then read from the working configuration table and converts that value, which is in minutes, to seconds. To find the zero point a 10 sample window is used. The first 9 samples are collected from the ADC at 1 second intervals, testing for the zero point will begin with the 10^{th} sample. Zero Calibration then begins to monitor for the zero point by taking samples and decrementing the calibration duration every second. If the calibration duration counter value becomes less than 1 the zero point was not found in the allotted time, and the Sensor Unit enters Fault Mode, issuing a Zero Calibration Fault. After taking a sample, that sample is checked to see if the zero point has been found. To perform the test for the zero point the slope of the first order polynomial least squares fit of the 10 data points is computed. If the slope is below the maximum calibration slope variation threshold then the zero point is found and the current reading is recorded as the zero point.

The estimated reference point and the slope of the line connecting the zero and reference points are found as previously described in Section 3.1.2.2. The calculated slope and the zero point are then tested against the manufacturer's parameters to ensure that a valid calibration has occurred. The SLOPE must be between the minimum and maximum slopes and the zero point must be less than or equal to the maximum allowable zero point. If any of these tests fail, Fault Mode is entered and a Zero Calibration Fault is issued.

```
Send Acknowledge Once to Acknowledge Entry Into Zero Calibration Mode
IF( Message Received AND Message == Zero Calibration Start Command ) THEN
        Send Acknowledge in Response to Zero Calibration Start Command
        Calibration Duration = Read Zero Duration From Working Configuration Table in EEPROM
        Collect the First 9 Samples
        Zero Point Found = FALSE

        WHILE( Zero Point Found == FALSE )
                IF( Time For Sample ) THEN
                        Current Sample = Read ADC
                        Slope = Slope From Least Squares Fit to Last 10 Samples
                        Slope = Absolute Value of Slope
                        IF( Slope < Maximum Calibration Slope Variation ) THEN
                                Zero Point Found = TRUE
                                ADC_ZERO_CAL_Z = Current Sample
                        END IF
                        Calibration Duration = Calibration Duration -1
                        IF( Calibration Duration <= 0 ) THEN
                                Enter Fault Mode Due To A Zero Calibration Fault
                        END IF
                END IF
        END WHILE

        Calculate ADC_CAL_Z Value
        PPM_CAL = Read PPM_Cal from Working Calibration Table in EEPROM
        SLOPE = (PPM_CAL - 0) / (ADC_CAL_Z - ADC_ZERO_CAL_Z )
        IF( (SLOPE > Maximum Slope) OR (SLOPE < Minimum Slope) ) THEN
                Fault Mode Due To A Zero Calibration Fault
        END IF
        IF( ADC_ZERO_CAL_Z > Maximum ADC_ZERO_CAL ) THEN
                Fault Mode Due To A Zero Calibration Fault
        END IF
        Copy Previous 1 Calibration Table to Previous 2 Calibration Table
        Copy Working Calibration Table to Previous 1 Table
        Write New Slope to Working Calibration Table
        Write ADC_ZERO_CAL_Z to Working Calibration Table
        Write ADC_CAL_Z to Working Calibration Table
        Read the Current Date From the Real-Time Clock
        Write Date to Working Calibration Table
        Send Zero Calibration Complete Ok to Area Monitor
        Wait for Area Monitor to Acknowledge Zero Calibration Complete Ok Message
        Enter Normal Operation Mode
END IF
```

Figure 4-26: Pseudocode of the Sensor Unit Zero Calibration code.

The calibration tables are then updated to reflect the new calibration values and maintain the required historical data. The previous 2 calibration table is overwritten with the previous 1 calibration table, and the previous 1 calibration table is overwritten by the working calibration table. The SLOPE, ADC_ZERO_CAL_Z, ADC_CAL_Z, and date of calibration, fields in the current working calibration table are updated with the current values. The Sensor Unit sends the zero calibration complete message to the Area Monitor, waits for the Area Monitor to acknowledge and then enters Normal Operation Mode.

4.15.2 Design of Area Monitor Zero Calibration

The Zero Calibration can be performed from the built-in interface and from the remote computer. The design of the software for the Zero Calibration performed using the built-in interface is presented first, and the design of the Zero Calibration routine when performed from the remote computer is presented second.

The user at the Area Monitor initiates the Keypad Zero Calibration Operation by selecting the zero calibration menu option. The Area Monitor will wait until the acknowledge message is received before entering Keypad Zero Calibration Mode. Figure 4-27 shows the pseudocode for Zero Calibration Mode performed using the built-in man-machine interface. The software for the Zero Calibration routine is modeled as a finite state machine with two states. The first state, Prompt Start Zero Calibration, prompts the user to start the zero calibration. The second state, Wait for Zero Calibration Complete, waits for the zero calibration complete message from the Sensor Unit.

```
ZCal_State = Current State within Configuration Routine (Global Variable)
KeyPress = Keypres detected by Keypad ISR from the Keypad (Global Variable)
Message = Message received from Sensor Unit RS-232 link

Display Zero Calibration Start Prompt
ZCal_State = Prompt Start Zero Calibration

IF ( ZCal_State == Prompt Start Zero Calibration ) THEN
        IF( KeyPress == ENTER ) THEN
                ZCal_State = Wait For Zero Calibration Complete
                Send Start Zero Calibration to Sensor Unit
                Wait for Acknowledge From Sensor Unit
        END IF
        KeyPress = NULL
ELSE IF( ZCal_State == Wait For Zero Calibration Complete ) THEN
        IF( ( Sensor Unit Message Received ) AND (Message == Zero Calibration Complete ) ) THEN
                Send Zero Calibration Complete to Remote Computer
                Send Acknowledge to Sensor Unit
                Set Top Level State to Normal Operation Mode
        END IF
END IF-ELSE
```

Figure 4-27: Pseudocode for Area Monitor Zero Calibration Mode using the built-in interface.

The Area Monitor initiates the Zero Calibration sending the zero calibration command to the Sensor Unit. The user is then prompted to start the Zero Calibration. The zero calibration start command is sent to the Sensor Unit after the user presses the enter key and the Area Monitor waits for the Sensor Unit to send the acknowledge message.

After receiving the acknowledge message the Zero Calibration State is updated to Wait for Zero Calibration Complete. The Area Monitor waits for the zero calibration complete message from the Sensor Unit, indicating the Zero Calibration has successfully completed. The Area Monitor RS-232 ISR receives the message and sets the Sensor Unit Message Received flag, storing the received byte in the Message variable. The zero calibration complete message is then sent to the remote computer and the acknowledge message is sent to the Sensor Unit. Finally, the Area Monitor enters Normal Operation Mode.

The design of the routine to perform the Zero Calibration from the remote computer is described below. The remote computer will initiate the Zero Calibration operation sending the zero calibration command to the Area Monitor. The remote computer RS-232 ISR will receive the zero calibration command, send the zero calibration command to the Sensor Unit and switch state to Remote Computer Zero Calibration Mode.

Figure 4-28 shows the pseudocode for the Zero Calibration Mode when performed from the remote computer. The 4-character display will display the zero calibration message, informing user that a zero calibration is currently taking place. Remote Computer Zero Calibration waits for the remote computer to send the start zero calibration command. Again, the remote computer RS-232 ISR handles the received message, setting the Remote Computer Message Received flag and storing the message in the Message variable. The Area Monitor will send the start zero calibration command to the Sensor Unit after receiving the start zero command from the remote computer, and waits for the Sensor Unit to acknowledge. The Area Monitor waits for the Sensor Unit to finish the zero calibration operation. The Sensor Unit sends the zero calibration complete message after completing the zero calibration. After the zero calibration complete message is received the Area Monitor sends the zero calibration complete message to the remote computer, sends the acknowledge message to the Sensor Unit and enters Normal Operation Mode.

```
ZCal_StatePC = Current State within Zero Calibration Routine (Global Variable)
R_Message = Message received from the Remote Computer
Display Zero Calibration Message
ZCal_StatePC = Wait For Start Zero Calibration

IF( ZCal_StatePC == Wait For Start Zero Calibration ) THEN
        IF( ( Remote Computer Message Received ) ) AND (R_Message == Zero Calibration Start Command ) ) THEN
                Send Start Zero Calibration to Sensor Unit
                Wait for Sensor Unit to Acknowledge
                ZCal_StatePC = Wait For Zero Calibration Complete
        END IF
ELSE IF( ZCal_StatePC == Wait For Zero Calibration Complete ) THEN
        IF(  Sensor Unit Message Received ) AND (R_Message == Zero Calibration Complete) ) THEN
                Send Zero Calibration Complete to the Remote Computer
                Send Acknowledge to Sensor Unit
                Set Top-Level State to Normal Operation Mode
        END IF
END IF-ELSE
```

Figure 4-28: Pseudocode for Area Monitor Zero Calibration performed from the remote computer.

4.16 DESIGN OF SPAN AND ZERO CALIBRATION MODE

4.16.1 Design of Sensor Unit Span and Zero Calibration Mode

As described in Section 3.1.2.1, the Span and Zero Calibration will obtain the zero and reference points, and using those two points, calculate the SLOPE. The system will enter Fault Mode if the zero point and/or the reference point are not observed within the specified zero duration or reference duration times respectively, or if the value of SLOPE calculated from the two points is not within the set limits. The Area Monitor initiates the Span and Zero Calibration in response to user action, sending the span calibration command to the Sensor Unit.

The software for Span and Zero Calibration routine is modeled as a finite state machine, with five states. The first state, Waiting for Zero Calibration Start Command, waits for the user to start the zero calibration. The next state, Perform Zero Calibration, monitors the sensor output and obtains the zero point. The third state, Receiving Data, waits until the Area Monitor sends the scale factor and reference PPM to the Sensor Unit. The fourth state, Waiting for Span Calibration Start Command, waits for the user to initiate the span calibration, and the final state, Perform Span Calibration, finds the reference point and updates the calibration data.

Figure 4-29 shows the pseudocode for Span and Zero Calibration Mode in the Sensor Unit. The Sensor Unit first waits to receive the start zero calibration command from the Area Monitor, before starting to monitor for the zero point. While waiting the Sensor Unit sends the acknowledge command to the Area Monitor. When the start zero calibration command is received, the Sensor Unit updates the Span Calibration State (SCalState) to Perform Zero Calibration, to monitor for the zero point. The process to determine when the zero and reference point has been observed is the same as the process to find the zero point described in Section 4.15.1.

After the zero point is found, the zero calibration complete message is sent to the Area Monitor and the Sensor Unit then waits for the acknowledge message to be received from the Area Monitor. After receiving the acknowledge message the Span Calibration State is updated to Receiving Data, to receive the scale factor and PPM_CAL from the Area Monitor. The Sensor Unit RS-232 ISR will receive the data, setting the All Span Calibration Data Received flag after the last byte has been received. The scale factor and the PPM_CAL are read from the temporary buffer and the Span Calibration State updated Waiting for Span Calibration Start Command, to wait for the user to initiate the span calibration.

```
IF( SCalState == Waiting For Zero Calibration Start Command ) THEN
        Send Acknowledge to Area Monitor
        IF( Area Monitor Message Received == TRUE ) THEN
                IF( Message == Zero Calibration Start Command ) THEN
                        SCalState = Perform Zero Calibration
                END IF
        END IF
ELSE IF( SCalState == Perform Zero Calibration ) THEN
        Zero Point Found = FALSE
        Calibration Duration = Read Zero Duration From Working Configuration Table in EEPROM
        Collect the First 9 Samples

        WHILE( Zero Point Found == FALSE )
                IF( Time For Sample ) THEN
                        Current Sample = Read ADC
                        Slope = Slope From Least Squares Fit to Last 10 Samples
                        Slope = Absolute Value of Slope
                        IF( Slope < Maximum Calibration Slope Variation ) THEN
                                Zero Point Found = TRUE
                                ADC_ZERO_CAL_SZ = Current Sample
                        END IF
                        Calibration Duration = Calibration Duration -1
                        IF( Calibration Duration <= 0 ) THEN
                                Enter Fault Mode Due To A Span Calibration Fault
                        END IF
                END IF
        END WHILE

        Send Zero Calibration Complete to the Area Monitor
        Wait for Acknowledge from Area Monitor
        SCalState = Receiving Data
ELSE IF( SCalState == Receiving Data ) THEN
        IF(All Span Calibration Data Received == TRUE ) THEN
                Scale Factor = Read Scale Factor from Temporary Buffer
                Gas Concentration = Read Gas Concentration from Temporary Buffer
                SCalState = Waiting For Span Calibration Start Command
        END IF
IF( SCalState == Waiting For Span Calibration Start Command ) THEN
        IF( Area Monitor Message Received == TRUE ) THEN
                IF( Message == Span Calibration Start Command ) THEN
                        SCalState = Perform Span Calibration
                END IF
        END IF
ELSE IF( SCalState == Perform Span Calibration ) THEN
        Span Point Found = FALSE
        Calibration Duration = Read Span Duration From Working Configuration Table in EEPROM
        Collect the First 9 Samples

        WHILE( Span Point Found == FALSE )
                IF( Time For Sample ) THEN
                        Current Sample = Read ADC
                        Slope = Slope From Least Squares Fit to Last 10 Samples
                        Slope = Absolute Value of Slope
                        IF( Slope < Maximum Calibration Slope Variation ) THEN
                                Zero Point Found = TRUE
                                ADC_CAL_SZ = Current Sample
                        END IF
                        Calibration Duration = Calibration Duration -1
                        IF( Calibration Duration <= 0 ) THEN
                                Enter Fault Mode Due To A Span Calibration Fault
                        END IF
                END IF
        END WHILE

        Send Span Calibration Complete to the Area Monitor
        Current Temperature = Read Temperature Sensor
        SLOPE = PPM_CAL / [(ADC_CAL_SZ - ADC_ZERO_CAL_SZ) * Scale Factor]
        IF( (SLOPE > Max_SLOPE) OR (SLOPE < Min_SLOPE) ) THEN
                Enter Fault Mode Due To A Span Calibration Fault
        END IF

        Copy Previous 1 Calibration Table to Previous 2 Calibration Table
        Copy Working Calibration Table to Previous 1 Table
        Write New Slope to Working Calibration Table
        ADC_ZERO_CAL_Z = ADC_ZERO_CAL_SZ
        ADC_CAL_Z = ADC_CAL_SZ
        Write ADC_ZERO_CAL_Z to Working Calibration Table
        Write ADC_CAL_Z to Working Calibration Table
        Write ADC_ZERO_CAL_SZ to Working Calibration Table
        Write ADC_CAL_SZ to Working Calibration Table
        Write Current Temperature to Working Calibration Table
        Write PPM_CAL to Working Calibration Table
        Write Scale Factor to Working Calibration Table
        Read the Current Date From the Real-Time Clock
        Write Date to Working Calibration Table
        Send Span Calibration Complete Ok to Area Monitor
        Wait for Acknowledge from Area Monitor
        Enter Normal Operation Mode
END ELSE-IF
```

Figure 4-29: Pseudocode for Span and Zero Calibration in the Sensor Unit.

When the start span calibration command is received, the Span Calibration State is updated to, Perform Span Calibration. The reference point is found using the same procedure as was used to find the zero point. The SLOPE is calculated as described in Section 3.1.2.1 using PPM_CAL, the zero point, and reference point. The SLOPE is the checked against the maximum and minimum allowable slope values. If the SLOPE is outside either value, Fault Mode is entered and a Span Calibration Fault is issued.

The calibration tables are updated to reflect the new calibration values and to maintain the required historical data. The previous 2 calibration table is overwritten with the previous 1 calibration table, and the previous 1 calibration table is overwritten by the working calibration table. The new calibration values are written into the working calibration table. The reference and zero points found are used to overwrite both sets of calibration points in the working calibration table. The span calibration complete message is sent to the Area Monitor and the Sensor Unit waits for the Area Monitor to reply with the acknowledge message. After the acknowledge message has been received the Sensor Unit enters Normal Operation Mode.

4.16.2 Design of Area Monitor Span and Zero Calibration Mode

Two functions will perform the Span and Zero Calibration in the Area Monitor. One function will perform the calibration when the built-in interface is used, and the second when the remote computer performs the calibration. The design of the routine to perform the calibration using the built-in interface is presented first, followed by the design of the routine for the remote computer.

The Area Monitor enters Span and Zero Calibration Mode after the user selects the span and zero calibration operation from the menu. The span calibration command is sent to the

Sensor Unit and the Area Monitor enters Keypad Span and Zero Calibration Mode. The code for the Keypad Span and Zero Calibration routine is modeled as a finite state machine with eight states.

Keypad Span and Zero Calibration Mode is entered when the user selects the span and zero calibration menu option. The Area Monitor will wait until the acknowledge message is received before entering Keypad Span and Zero Calibration Mode. Figure 4-30 shows the pseudocode for Keypad Span and Zero Calibration Mode performed using the built-in interface. The Area Monitor prompts the user to start the zero calibration, displaying the start zero calibration prompt message on the 4-character display. The Area Monitor remains in this state until the enter key is pressed. When the enter key is pressed the start zero calibration command is sent to the Sensor Unit, instructing the Sensor Unit to find the zero point and the span calibration state is updated to wait for the zero calibration complete message.

The Area Monitor RS-232 ISR handles the receipt of messages from the Sensor Unit, sets the Sensor Unit Message Received flag, and stores the received byte in the Message variable when a byte is received. The zero calibration complete message received after the Sensor Unit has found the zero point triggers the update of the span calibration state to prompt the user to enter the scale factor. The Area Monitor sends the acknowledge message to the Sensor Unit when entering the prompt the user to enter the scale factor state. The user must press the enter key to enter the scale factor value. The user will use the up, down, left, and right keys to alter the value. The scale factor is stored when the enter key is pressed. The scale factor is value between 0 and 255 and is interpreted as a percentage.

```
SCal_State = Current State within Configuration Routine (Global Variable)
Data = Data collected from user, updated by the keypad ISR (Global Variable)
KeyPress = Keypres detected by Keypad ISR from the Keypad (Global Variable)
R_Message = Message received from the Remote Computer

Display Zero Calibration Start Prompt
SCal_State = Wait For Start Zero Calibration

IF( SCal_State == Wait For Start Zero Calibration ) THEN
        IF( KeyPress == ENTER ) THEN
                SCal_State = Wait For Zero Calibration Complete
                Send Start Zero Calibration to Sensor Unit
        END IF
        KeyPress = NULL
ELSE IF( SCal_State == Wait For Zero Calibration Complete ) THEN
        IF(Sensor Unit Message Received ) AND (Message == Zero Calibration Complete) ) THEN
                SCal_State = Prompt For Scale Factor
                Send Zero Calibration Complete to Remote Computer
        END IF
ELSE IF( SCal_State ==Prompt For Scale Factor ) THEN
        Display Scale Factor Prompt
        IF( KeyPress == ENTER ) THEN
                SCal_State = Enter Scale Factor
                Data = Read Scale Factor from Working Calibration Table in EEPROM
        END IF
        KeyPress = NULL
ELSE IF( SCal_State == Enter Scale Factor ) THEN
        Send Acknowledge to Sensor Unit
        Display Data to User
        IF( KeyPress == ENTER ) THEN
                Scale_Factor = Data
                SCal_State = Prompt For Reference Concentration
                Write Scale Factor to Working Calibration Table
                KeyPress = NULL
        END IF
ELSE IF( SCal_State == Prompt For Reference Concentration ) THEN
        Display Reference Concentration Prompt
        IF( KeyPress == ENTER ) THEN
                SCal_State = Enter Gas Concentration
                Data = Read Reference Calibration from Working Calibration Table in EEPROM
        END IF
        KeyPress = NULL
ELSE IF( SCal_State == Enter Reference Concentration ) THEN
        Display Data to User
        IF( KeyPress == ENTER ) THEN
                IF( Data <= ( Maximum Sensor Range / (Scale Factor / 100) ) ) THEN
                        Write Reference Concentration to Working Calibration Table
                        SCal_State = Prompt To Start Span Calibration
                END IF
                KeyPress = NULL
        END IF
ELSE IF( SCal_State == Prompt To Start Span Calibration ) THEN
        Display Span Calibration Start Prompt
        IF( KeyPress == ENTER ) THEN
                SCal_State = Send Data
        END IF
        KeyPress = NULL
ELSE IF( SCal_State == Send Data ) THEN
        Send Reference Concentration to the Sensor Unit
        Send  Scale_Factor to the Sensor Unit
        Send Start Span Calibration Command to Sensor Unit
        SCal_State = Wait For Span Calibration Complete
ELSE IF( SCal_State == Wait For Span Calibration Complete  ) THEN
        IF( ( Sensor Unit Message Received ) AND ( Message Received == Span Calibration Complete ) ) THEN
                Send Span Calibration Complete to the Remote Computer
                Send Acknowledge to Sensor Unit
                Set Top-Level State to Normal Operation Mode
        END IF
END IF-ELSE
```

Figure 4-30: Pseudocode for Span and Zero Calibration at the Area Monitor using the built-in interface.

After the enter key is pressed the span calibration state is updated to prompt for the reference calibration. The user presses enter, updating the span calibration state to enter the reference concentration. The user enters the reference concentration identically to the scale factor. When the user presses the enter key the reference calibration is checked to be within the limit specified by the manufacturer. The reference concentration can be no greater than the maximum range of the sensor divided by the scale factor as a percentage. After a valid reference concentration is entered, the span calibration state is updated, prompting the user to start the span calibration.

When the user presses the enter key the start span calibration command is sent to the Sensor Unit, instructing the Sensor Unit to find the reference point. The span calibration state is then updated to wait for the span calibration complete message to be received from the Sensor Unit. After the span calibration complete message is received, the span calibration complete message is sent to the remote computer and the Area Monitor sends the acknowledge message to the Sensor Unit. After sending the acknowledge message the top-level state is set to Normal Operation Mode.

The Span and Zero Calibration can be performed from the remote computer in addition to the built-in interface. The remote computer will initiate the span and zero calibration by sending the span calibration command to the Area Monitor. The remote computer RS-232 ISR receives and handles the span calibration command, switching state to perform the Zero and Span Calibration. The Remote Computer Span and Zero Calibration routine is modeled as a finite state machine with five states.

Figure 4-31 shows the pseudocode for the Remote Computer Span and Zero Calibration when performed from the remote computer. The Area Monitor updates the 4-character display to indicate that a span and zero calibration is being performed and waits to receive the start zero calibration command. After receiving the start zero calibration command, the Area Monitor sends the start zero calibration command to the Sensor Unit and updates the span calibration state to wait for the zero calibration complete message from the Sensor Unit.

```
SCal_StatePC = Current State within Span and Zero Calibration Routine (Global Variable)
R_Message = Message received from the Remote Computer

Display Performing Span and Zero Calibration Message
SCal_StatePC = Wait For Start Zero Calibration Command

IF( SCal_StatePC == Wait For Start Zero Calibration Command ) THEN
        IF( ( Remote Computer Message Received ) AND (R_Message == Start Zero Calibration Command) ) THEN
                Send Start Zero Calibration Command to Sensor Unit
                SCal_StatePC = Wait For Zero Calibration Complete
        END IF
ELSE IF( SCal_StatePC == Wait For Zero Calibration Complete ) THEN
        IF( ( Remote Computer Message Received ) AND ( R_Message== Zero Calibration Complete ) ) THEN
                Send Zero Calibration Complete to the Remote Computer
                SCal_StatePC = Receive Data
        END IF
ELSE IF( SCal_StatePC == Receive Data ) THEN
        Send Acknowledge to Sensor Unit
        IF( All Span Calibration Data Received ) THEN
                Convert Reference Calibration to Microchip Float-Point Format
                Write Reference Calibration to Working Calibration Table
                Write Scale Factor to Working Calibration Table
                SCal_StatePC = Wait For Start Span Calibration
        END IF
ELSE IF( SCal_StatePC == Wait For Start Span Calibration ) THEN
        IF( ( Remote Computer Message Received ) AND (R_Message == Start Span Calibration Command) ) THEN
                Send Start Span Calibration Command to Sensor Unit
                SCal_StatePC = Wait For Span Calibration Complete
        END IF
ELSE IF( SCal_StatePC == Wait For Span Calibration Complete ) THEN
        IF( ( Sensor Unit Message Received ) AND (R_Message == Span Calibration Complete) ) THEN
                Send Span Calibration Complete to the Remote Computer
                Send Acknowledge to Sensor Unit
                Set Top-Level State to Normal Operation Mode
        END IF
END IF-ELSE
```

Figure 4-31: Pseudocode for Span and Zero Calibration performed from the remote computer.

After receiving the zero calibration complete message from the Sensor Unit, the Area Monitor sends that message to the remote computer, sends the acknowledge message to the

Sensor Unit, and updates the span calibration state to receive the scale factor and the reference concentration. The Remote Computer RS-232 ISR receives the data setting the All Span Calibration Data Received flag after the last byte of data is received. The Area Monitor converts the reference concentration to Microchip floating-point format. The converted reference concentration and scale factor are written to the current calibration table and are then sent to the Sensor Unit. The remote computer must send the start span calibration command to instruct the system to find the reference point.

When the start span calibration command is received, the Area Monitor sends the start span calibration command to the Sensor Unit and updates the span calibration state to wait for the span calibration to complete. The Sensor Unit will send the span calibration complete to the Area Monitor after the Span and Zero Calibration is completed. The Area Monitor after receiving span calibration complete message sends the span calibration complete to the remote computer and sends the acknowledge message to the Sensor Unit. The Area Monitor then enters Normal Operation Mode.

4.17 DESIGN OF UPDATE LIMITS MODE

The Update Limits routine, as previously described in Section 3.1.5, allows the user to change system settings specific to the purpose or location of the system. The Update Limits routine can only be executed from the remote computer and will update the altitude, the latching and acknowledge alarm settings, and the system time and date. First, the design of the code for the Update Limits Mode for the Sensor Unit is presented, followed by the design of the code for the Area Monitor.

4.17.1 Design of Sensor Unit Update Limits Mode

The Sensor Unit enters Update Limits Mode after receiving the update limits command from the Area Monitor. Figure 4-32 shows the pseudocode for Update Limits Mode. The Sensor Unit RS-232 ISR receives and collects the limits data sent from the Area Monitor to the Sensor Unit. A total of eight bytes of data will be received from the Area Monitor. After receiving the last byte of data, the Sensor Unit RS-232 ISR will set the All Limits Data Received flag. In response to the set flag, the software writes the altitude, the latching and acknowledge byte, the new date, and time to the Limits Table in EEPROM. A real-time clock maintains the system time and date. The date and time in the real-time clock chip is updated with the new date and time, sent as part of the limits data.

The real-time clock interfaces with the processor using an Intel style interface. The real-time clock stores the minute, hour, day, month, and year internally in individual registers. The address of the register to write and the data to write must be provided using the protocol. The registers are written by first triggering the address write phase, then writing the address of the desired register to the data lines, and waiting 100ns. The data write sequence of the protocol is then triggered and the data is written to the data lines. The real-time clock must be set to the update mode before a write to the time and date keeping registers can occur. The software first writes register B with the proper code to enable writing to the time and date registers. Each register is written to update the date and time. After all date and time registers are updated register B is written with the value to disable writing of the date and time registers and enabling normal time keeping operations. After the time and date are updated, the update limits complete message is sent to the Area Monitor and the Sensor Unit enters Normal Operation Mode.

```
IF( All Limits Data Received ) THEN
Write Altitude to Limits Table in EEPROM
Write Latching and Acknowledge Byte to Limits Table in EEPROM
Write Time and Date to Limits Table
Update Time and Date of Real-Time Clock

Send Update Limits Complete to Area Monitor
Set Top-Level State to Normal Operation Mode
END IF
```

Figure 4-32: Pseudocode for Update Limits Mode in the Sensor Unit.

4.17.2 Design of Area Monitor Update Limits Mode

The remote computer interface must be used to perform the Update Limits routine. The remote computer will initiate the Update Limits procedure, sending the update limits command to the Area Monitor. The remote computer RS-232 ISR receives the update limits command and causes the Area Monitor to enter Update Limits Mode.

Figure 4-33 shows the pseudocode for the Area Monitor Update Limits Mode. The Remote Computer RS-232 ISR receives and collects the limits data sent from the remote computer. The remote computer RS-232 ISR will set the All Limits Data Received flag after receiving the last byte of limits data from the remote computer. A total of eight bytes of data will be received from the remote computer and sent to the Sensor Unit. The Update Limits software, in response to the set flag, writes the altitude and latching and acknowledge settings to the Limits Table in EEPROM.

The update limits command is sent to the Sensor Unit followed by the limits data. After sending the limits data the All Limits Data Received flag is cleared and the Area Monitor waits for the Sensor Unit to complete the update limits routine. The Sensor Unit will send the update limits complete message to the Area Monitor after completing the update limits routine. The Area Monitor then sends the update limits message to the remote computer and enters Normal Operation Mode after receiving the update limits complete message.

```
IF ( All Limits Data Received ) THEN
Write Altitude to Limits Table in EEPROM
Write Latching and Acknowledge Byte to Limits Table in EEPROM

Send Update Limits Command to Sensor Unit

Send Altitude to Sensor Unit
Send Latching and Acknowledge Byte to Sensor Unit
Send Hour to Sensor Unit
Send Minute to Sensor Unit
Send Day to Sensor Unit
Send Month to Sensor Unit
Send Year to Sensor Unit
Clear All Limits Data Received Flag

ELSE
IF ( Sensor Unit Message Received ) AND ( Message == Update Limits Complete ) THEN
Send Update Limits Complete to Remote Computer
Set Top-Level State to Normal Operation Mode
END IF

END IF
```

Figure 4-33: Pseudocode for Update Limits Mode at the Area Monitor.

4.18 DESIGN OF DOWNLOAD SYSTEM DATA MODE

As previously described in Section 3.1.9, the remote computer requires a copy of the system data for operation and for inspection of the system operational history. The Sensor Unit will maintain the operational data and the historical system data. The data will be sent to the remote computer upon request. The design of the software for the Download System Data Mode for the Sensor Unit is described, followed by the design of the Area Monitor software.

4.18.1 Design of Sensor Unit Download System Data Mode

The Sensor Unit sends the data to the Area Monitor in response to the download system data command sent from the Area Monitor. Figure 4-34 shows the pseudocode for the Download System Data Mode. The Sensor Unit will first convert all floating-point values from the Microchip floating-point format to the IEEE 754 floating-point format. All eight tables stored in the Sensor Unit EEPROM are sent to the Area Monitor. The tables are sent to the Area Monitor in the following order: working configuration table, working calibration table, limits table, previous 1 configuration table, previous 1 calibration table, previous 2 configuration table, previous 2 calibration table, and finally the sensor data table. After sending all eight tables, the Sensor Unit enters Normal Operation Mode.

```
Convert All Floating-Point Values to IEEE 754 Floating-Point Format

Send Working Configuration Table to Area Monitor
Send Working Calibration Table to Area Monitor
Send Limits Table to Area Monitor
Send Previous 1 Configuration Table to Area Monitor
Send Previous 1 Calibration Table to Area Monitor
Send Previous 2 Configuration Table to the Area Monitor
Send Previous 2 Calibration Table to the Area Monitor
Send Sensor Data Table to the Area Monitor

Set Top-Level State to Normal Operations Mode
```

Figure 4-34: Pseudocode for the Sensor Unit Download System Data Mode.

4.18.2 Design of Area Monitor Download System Data Mode

The remote computer will initiate the download system data operation, sending the download system data command to the Area Monitor. The remote computer RS-232 ISR will receive and handle the download system data command, sending the download system data command to the Sensor Unit, and causes the Area Monitor to switch from Normal Operation Mode to Download System Data Mode.

Figure 4-35 shows the pseudocode for Download System Data Mode. The Sensor Unit will send all operational and historical data to the Area Monitor. The Area Monitor RS-232 ISR receives and stores the data from the Sensor Unit in a temporary buffer, setting the All Download Data Received flag after receiving the last byte of data. After the flag is set, the Area Monitor sends the data in the temporary buffer to the remote computer and enters Normal Operation Mode.

```
IF (All Download Data Received ) THEN
        Send All Data in Temporary Buffer to the Remote Computer
        Set Top-Level State to Normal Operation Mode
END IF
```

Figure 4-35: Pseudocode for the Sensor Unit Download System Data Mode.

4.19 DESIGN OF REQUEST SYSTEM STATE ROUTINE

As described in Section 3.1.10, the system will provide the remote computer with the ability to request the current operating state of the system. This section describes the design of the routine to handle the system state request. The request system state routine involves only the Area Monitor and the remote computer. The Sensor Unit does not participate in the request system state routine.

Figure 4-36 shows the pseudocode for the Request System State functionality in the Area Monitor. The Area Monitor maintains global variables indicating the current top-level state, and the current inner states of each routine containing inner state. The Area Monitor enters the request system state routine after receiving the request system state command from the remote computer. The Area Monitor sends two data bytes to the remote computer in response to the request system state command. The first byte will contain the current top-level state of the system. If the current top-level state is Zero Calibration, Span and Zero Calibration, Configuration, or Diagnostic, Sensor Warm-Up, the second byte sent is the inner state of the current operation. If the top-level state is Fault Mode, the outstanding fault is sent, and if the top-level state is Normal Operation, Alarm Mode, Download System Data, or Update Limits the second byte sent is a ‘0’.

```
Send Top-Level State to Remote Computer
IF( Top-Level State == Normal Operation Mode ) THEN
        Send '0' to Remote Computer
ELSE IF( Top-Level State == Alarm Mode ) THEN
        Send '0' to Remote Computer
ELSE IF( Top-Level State == Remote Computer Configuration Mode ) THEN
        Send Configuration State to Remote Computer
ELSE IF( Top-Level State == Keypad Configuration Mode ) THEN
        Send Configuration State to Remote Computer
ELSE IF( Top-Level State == Remote Computer Zero Calibration Mode ) THEN
        Send Zero Calibration State to Remote Computer
ELSE IF( Top-Level State == Keypad Zero Calibration Mode ) THEN
        Send Zero Calibration State to Remote Computer
ELSE IF( Top-Level State == Remote Computer Span Calibration ) THEN
        Send Span Calibration State to Remote Computer
ELSE IF( Top-Level State == Remote Computer Span Calibration ) THEN
        Send Span Calibration State to Remote Computer
ELSE IF( Top-Level State == Warm-Up Mode ) THEN
        Send Warm-Up State to Remote Computer
ELSE IF( Top-Level State == Download System Data ) THEN
        Send '0' to Remote Computer
ELSE IF( Top-Level State == Fault Mode ) THEN
        Send Outstanding Fault to Remote Computer
ELSE IF( Top-Level State == Update Limits Mode )
        Send '0' to Remote Computer
ELSE IF (Top-Level State == Diagnostic Mode)
        Send Diagnostic State to Remote Computer
ELSE IF( Top-Level State== Power-Up Mode )
        Send '0' to Remote Computer
END IF-ELSE
```

Figure 4-36: Pseudocode for Request System State in the Area Monitor.

4.20 DESIGN OF FAULT MODE

4.20.1 Design of Sensor Unit Fault Mode

The Sensor Unit enters Fault Mode after a fault has been detected. Section 3.1.12 describes the operation required to clear each fault. The Sensor Unit will perform the required action in response to user action. Figure 4-37 shows the pseudocode for Fault Mode in the Sensor Unit. For a zero calibration fault, span calibration fault, diagnostic fault, or a loss of power fault the Area Monitor must send the command to initiate the operation to clear the fault. The operation will successfully complete, clearing the fault, or will fail causing a return to Fault Mode.

The Sensor Unit RS-232 ISR will handle all incoming messages and switches state accordingly if the command initiates the operation required to clear the outstanding fault. Commands initiating operations other than the required operation are ignored. A zero calibration fault and span calibration fault require the Zero Calibration and Span and Zero Calibration operations respectively to successfully complete to clear the fault. A diagnostic fault requires that a successful diagnostic be performed to clear the fault.

For a Sensor Module Past Lifetime Fault the Sensor Module must be replaced. When the Sensor Module is removed the Sensor Unit will enter Sensor Module Missing Fault Mode. If the Sensor Module is missing, the Sensor Unit will continuously check if a new module is inserted. When the new Sensor Module is inserted, the Sensor Unit sends the warm-up message to the Area Monitor and enters Warm-Up Mode. If no action has been taken to clear the outstanding fault, the Sensor Unit will send the fault message identifying the outstanding fault to the Area Monitor every time through the main software loop.

```
IF( Area Monitor Message Received ) THEN
        Area Monitor Message Received = FALSE

        IF ( Outstanding Fault == Zero Calibration Fault ) THEN
                IF ( Message == Zero Calibration Command ) THEN
                        Set Top-Level State to Zero Only Calibration Mode
                END IF
        ELSE IF( Outstanding Fault== Span and Zero Calibration ) THEN
                IF ( Message == Span Calibration Command ) THEN
                        Set Top Level State to Span and Zero Calibration Mode
                END IF
        ELSE IF( Outstanding Fault== Diagnostic Failure ) THEN
                IF ( Message == Diagnostic Start Command ) THEN
                        Set Top-Level State to Diagnostic Mode
                END IF
        ELSE IF( Outstanding Fault== Loss of Power Fault ) THEN
                IF ( Message == Acknowledge Loss of Power ) THEN
                        Clear Writing EEPROM Flag
                        Send Acknowledge to the Area Monitor
                        Wait for Acknowledge from Area Monitor
                        Set Top-Level State to Warm-Up Mode
                        Send Warm-Up Message to Area Monitor
                END IF
        END IF-ELSE

ELSE IF ( Outstanding Fault == Sensor Module Missing Fault ) THEN
        IF ( Sensor Module Present == TRUE )
                Set Top-Level State to Warm-Up Mode
                Send Warm-Up Message To Area Monitor
        ELSE
                Send Sensor Module Missing Fault Message to Area Monitor
        END IF
ELSE IF ( Outstanding Fault == Zero Calibration Fault ) THEN
        Send Zero Calibration Fault Message to Area Monitor
ELSE IF ( Outstanding Fault == Span and Zero Calibration ) THEN
        Send Span Calibration Fault Message to Area Monitor
ELSE IF ( Outstanding Fault == Diagnostic Failure ) THEN
        Send Diagnostic Fault Message to Area Monitor
ELSE IF ( Outstanding Fault == Loss of Power Fault ) THEN
        Send Loss of Power Fault Message to Area Monitor
ELSE IF ( Outstanding Fault == Sensor Module Past Specified Lifetime Fault ) THEN
        Send Sensor Module Past Specified Lifetime Fault Message to Area Monitor
END IF-ELSE
```

Figure 4-37: Pseudocode for Fault Mode in the Sensor Unit.

For a loss of power fault, after receiving the acknowledge loss of power message, the Sensor Unit will send the acknowledge message to the Area Monitor. The Sensor Unit will then wait for the acknowledge message from the Area Monitor.

4.20.2 Design of Area Monitor Fault Mode

Fault Mode will inform the user of the outstanding fault. The Area Monitor display will blink displaying the fault code in red, and will send the fault message to the remote computer. The user interface will provide the user the ability to perform the required action to clear the fault.

Fault Mode is entered in response to a fault message from the Sensor Unit. The Area Monitor will initiate the operation required to clear the outstanding fault based on user input. Figure 4-38 shows the pseudocode for Fault Mode in the Area Monitor. The Area Monitor sends the message identifying the current fault to the remote computer and displays the current fault message on the 4-character display. The user must perform the action specified in Section 3.1.12 to clear the outstanding fault. If the operation succeeds, the system will leave Fault Mode, otherwise the operation will fail, and the software will return to Fault Mode.

The remote computer can initiate the action required to clear the outstanding fault. The remote computer RS-232 ISR will set the remote computer message received flag when a message is received from the remote computer. All messages received from the remote computer are checked against the outstanding fault, in the Fault Mode routine. If the command initiates the action required to clear the outstanding fault, the Area Monitor sends that command to the Sensor Unit and enters the appropriate top-level state to perform that action. The Area Monitor waits to receive the acknowledge message from the Sensor Unit, after sending the acknowledge loss of power fault message to the Sensor Unit. If the command does not initiate the required action the Area Monitor relies sending the deny message, informing the remote computer that the requested operation will not be performed.

```
IF( Outstanding Fault == Zero Calibration Fault ) THEN
        Send Zero Calibration Fault Message to Remote Computer
        Display Zero Calibration Fault Message
ELSE IF( Outstanding Fault == Span and Zero Calibration ) THEN
        Send Span Calibration Fault Message to Remote Computer
        Display Span Calibration Fault Message
ELSE IF( Outstanding Fault == Diagnostic Failure ) THEN
        Send Diagnostic Fault Message to Remote Computer
        Display Diagnostic Fault Message
ELSE IF( Outstanding Fault == Loss of Power Fault ) THEN
        Send Loss of Power Fault Message to Remote Computer
        Display Loss of Power Fault Message
ELSE IF( Outstanding Fault == Sensor Module Past Specified Lifetime Fault ) THEN
        Send Sensor Module Past Specified Lifetime Fault Message to Remote Computer
        Display Sensor Module Past Lifetime Fault Message
IF( Outstanding Fault == Sensor Module Missing Fault ) THEN
        IF( Sensor Module Present == TRUE )
                Set Top-Level State to Warm-Up Mode
                Send Warm-Up Message To Remote Computer
        END IF
ELSE IF( Remote Computer Message Received ) THEN
        Remote Computer Message Received = FALSE

        IF( Outstanding Fault == Zero Calibration Fault ) THEN
                IF( Remote Computer Message == Zero Calibration Command ) THEN
                        Zero Calibration State = Wait For Zero Calibration Start Command
                        Set Top-Level State to Remote Computer Zero Only Calibration Mode
                        Send Zero Calibration Command to Sensor Unit
                ELSE
                        Send Deny Message to Remote Computer
                END IF
        ELSE IF( Outstanding Fault == Span and Zero Calibration ) THEN
                IF( Remote Computer Message == Span Calibration Command ) THEN
                        Send Span Calibration Command to Sensor Unit
                        Span Calibration State = Wait For Zero Calibration Start Command
                        Set Top Level State to Remote Computer Span and Zero Calibration Mode
                ELSE
                        Send Deny Message to Remote Computer
                END IF
        ELSE IF( Outstanding Fault == Diagnostic Failure ) THEN
                IF( Remote Computer Message == Diagnostic Start Command ) THEN
                        Diagnostic State = Wait For Area Monitor Test Command
                        Send Diagnostic Start Command to Sensor Unit
                        Set Top-Level State to Diagnostic Mode
                ELSE
                        Send Deny Message to Remote Computer
                END IF
        ELSE IF( Outstanding Fault == Loss of Power Fault ) THEN
                IF( Remote Computer Message == Acknowledge Loss of Power ) THEN
                        Temporary Buffer Write Address = 0
                        Warm-Up State = Receive New Sensor Module Data
                        Set Top-Level State to P-Up Mode
                        Send Acknowledge Loss of Power Message to Sensor Unit
                        Wait for Acknowledge From Sensor Unit
                ELSE
                        Send Deny Message to Remote Computer
                END IF
        END IF-ELSE

ELSE IF( KeyPress == Enter ) THEN
        KeyPress = NULL

        IF( Outstanding Fault == Zero Calibration Fault ) THEN
                Zero Calibration State = Wait For Zero Calibration Start Command
                Set Top-Level State to Keypad Zero Only Calibration Mode
                Send Zero Calibration Command to Sensor Unit
                Send Zero Calibration Command to Remote Computer
        ELSE IF( Outstanding Fault == Span and Zero Calibration ) THEN
                Send Span Calibration Command to Sensor Unit
                Send Span Calibration Command to Remote Computer
                Span Calibration State = Wait For Zero Calibration Start Command
                Set Top Level State to Remote Computer Span and Zero Calibration Mode
        ELSE IF( Outstanding Fault == Diagnostic Failure ) THEN
                Diagnostic State = Wait For Area Monitor Test Command
                Send Diagnostic Start Command to Sensor Unit
                Send Diagnostic Start Command to Remote Computer
                Set Top-Level State to Diagnostic Mode
        ELSE IF( Outstanding Fault == Loss of Power Fault ) THEN
                Temporary Buffer Write Address = 0
                Warm-Up State = Receive New Sensor Module Data
                Set Top-Level State to P-Up Mode
                Send Acknowledge Loss of Power Message to Remote Computer
                Send Acknowledge Loss of Power Message to Sensor Unit
                Wait for Acknowledge From Sensor Unit
        END IF-ELSE

END IF-ELSE
```

Figure 4-38: Pseudocode for Fault Mode in the Area Monitor.

The user can also clear the outstanding fault by pressing the enter key on the keypad to initiate the operation required to clear the fault. The Area Monitor sends the command to initiate the required action to the Sensor Unit and to the remote computer, sets all required flags and state variables, and then enters the top-level state to perform the required operation.

4.21 DESIGN OF THE MENU INTERFACE

The design of the menu interface follows from the requirements and specifications in Section 3.1.11. The menus display the currently available options to the user. The 4-character display in the Area Monitor displays the menus. Four character codes representing the options and system status are used to provide the menu system. The 4-character strings used in the system are shown in Table 1.

Two display drivers control the 4-character display, each controlling two of the four characters. The display drivers communicate with the processor using the I^2C bus protocol and contain internal font tables. The processor provides the code corresponding to the character to display in the font table and the display driver asserts the signals required to display that character.

A software display driver handles the translation and sending of the menu string to the display drivers for display on the 4-character display. Figure 4-39 shows the flowchart for the display driver. The display driver will read a character string of four characters from a global menu string buffer. The font code for each character in the global string buffer is determined and stored in a local buffer. After determining the font codes for all four characters the font codes are sent to the display drivers.

Table 1: 4-Character strings displayed.

Menu Display	**4-Character String**
Configuration Menu	CONF
Zero Only Calibration Menu	ZOCA
Span Calibration Menu	ZASC
Diagnostic Menu	DIAG
Diagnostic Fault	DFLT
Sensor Missing Fault	MFLT
Sensor Past Lifetime Fault	LFLT
Zero Calibration Fault	ZFLT
Span Calibration Fault	SFLT
Loss of Power Fault	WFLT
Warm-Up Mode Indicator	WARM
Power-Up Mode Indicator	PWER
Download System Data Indicator	DWLD
Update Limits Indicator	ULIM
Waiting for Zero Calibration Start	ZRDY
Waiting for Reference Calibration Start	SRDY
Prompt to Enter Reference PPM	GCON
Prompt to Enter Scale Factor	SCFA
Prompt to Enter High Alarm Threshold	HIAL
Prompt to Enter Low Alarm Threshold	LOAL
Prompt to Enter Zero Calibration Duration	ZCDU
Prompt to Enter Reference Calibration Duration	RCDU
PPM Reading Over-Range	OVER

The I^2C bus requires each device attached to the bus to have a unique address, the two display drivers have unique addresses. The processor first sends the address of the display driver controlling the leftmost two characters. The display driver contains registers that hold the character to display, different colors are displayed by writing the character code to different registers. The address of the register to write is then sent to the display driver, followed by the character code to write. The process is repeated for the display driver controlling the rightmost two characters.

The Area Monitor must display numeric data as well. The PPM, and data entries for configuration and calibration must be displayed to the user. A software routine converts numerical data, such as integer and floating-point values, into 4-character strings to be displayed, using the routines discussed above.

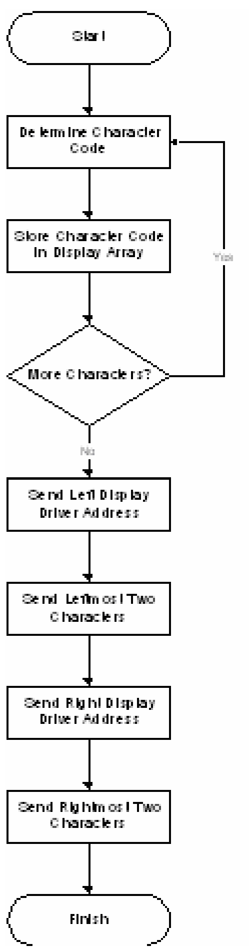

Figure 4-39: Flowchart for the display driver.

The value of the number determines the location of the decimal point in the 4-character string, providing maximum precision for the value to be displayed. In a fixed decimal point system, large values may be truncated. For instance 100 may appear as 10.0. Increaseing the range of the display by moving the decimal point to the right reduces the precision to which the instrument can display readings. Basing the position of the decimal on the value of the number to be displayed customizes the display to provide the entire number in the greatest possible precision.

Figure 4-40 shows the pseudocode for the routine to convert integer and floating-point data to 4-character strings for the display driver routine. The position of the decimal point is determined as follows. If the number is less than 1 then the decimal point is leftmost character. The three digits to the right of the decimal point are determined from value. The decimal point is second character from the left if the number is between 1 and 10. The digit to the left of the decimal point and the two digits to the right of the decimal point are determined from value. The decimal point is the third character if the number is between 10 and 100. The two digits to the left of the decimal point and the digit to the right of the decimal point are determined from value. The decimal point is the fourth character from the left if the number is between 100 and 1000. The three digits to the left of the decimal point are determined from value.

```
IF( value < 1 ) THEN
        Decimal Point is Leftmost character
        Convert the Three Digits Immediately Right of the Decimal Into Characters
ELSE IF( value < 10 ) THEN
        Decimal Point is 2nd Character From the Left
        Convert Digit to the Left of the Decimal Point To a Character
        Convert the Two Digits to the Right of the Decimal Point Into Characters
ELSE IF( value < 100 ) THEN
        Decimal Point is 3rd Character from the Left
        Convert the Two Digits to the Left of the Decimal Point Into Characters
        Convert the Digit to the Right of the Decimal Point To a Character
ELSE IF( value < 1000 ) THEN
        Decimal Point is 4th Character from the Left
        Convert the Three Digits to the Left of the Decimal Point Into Characters
END ELSE-IF
```

Figure 4-40: Pseudocode of routine converting integer and floating point values into 4-character strings.

4.22 DESIGN OF NON-VOLATILE MEMORY SPACE

Both the Area Monitor and the Sensor Unit must maintain several sets of data to perform the operations discussed in the preceding sections. The specifications and requirements previously described in Section 3.1.8 require the system to store operational data, historic data, and to maintain that data through power loss. The PIC processors used in the system contain a limited amount of on-board EEPROM, which provides the non-volatile storage for the data. As previously described, the operational data in the system is divided into multiple tables. Each table contains information required for carrying out specific functions in the system.

The Sensor Unit contains four tables containing working data, and four tables containing historic data. Figure 4-41 shows the layout of the individual data tables in the Sensor Unit EEPROM. The EEPROM will contain eight data tables and one byte is used as the writing EEPROM flag. The writing EEPROM flag is used to determine if power was lost while updating one or more of the tables in EEPROM. This flag is used to determine if a loss of power fault occurred when the system is in Power-Up Mode. The writing EEPROM flag is written immediately before writing data to the EEPROM and is clear after the last byte has been written to the EEPROM.

The Area Monitor maintains the same four working tables as the Sensor Unit. The four working tables provide data for the man-machine interface, and settings to control the alarms and external relays. The Area Monitor employs the same data layout in EEPROM, except that the previous calibration and configuration tables contain no data.

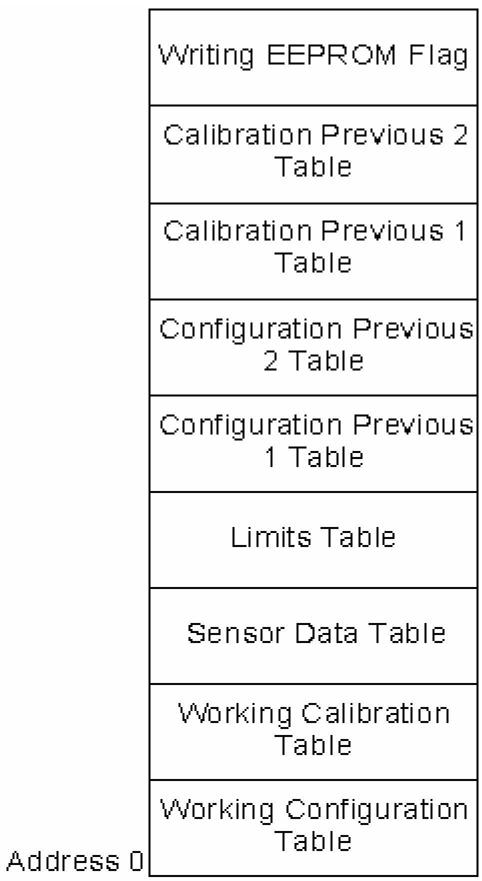

Figure 4-41: Top-Level layout of the data tables in the Sensor Unit EEPROM.

4.22.1 Design of Configuration Table

The configuration table contains all information collected during a configuration. Specifically the configuration table contains four values, the low alarm threshold, the high alarm threshold, the zero calibration duration, and the reference calibration duration. The low and high alarm thresholds are in units of PPM and are used to determine the alarm status after the PPM has been computed. The zero and reference calibration durations are in units of minutes and are used during calibrations to determine how long to wait for the observation of a valid calibration point. Table 2 shows the layout of the configuration data within the configuration table.

Table 2: Sensor Unit configuration data table layout.

Offset From Table Start	Table Element Name	Software Data Type
0	Low Alarm Threshold	Microchip Floating-Point
4	High Alarm Threshold	Microchip Floating-Point
8	Zero Calibration Duration	8-Bit Integer
9	Reference Calibration Duration	8-Bit Integer

4.22.2 Design of Calibration Table

The calibration table contains all information required to perform a calibration and all information calculated during a calibration. The calibration table contains nine values. The calibration table contains the zero and reference calibration points found during a Zero Calibration and a Span and Zero Calibration, the temperature at the time of the calibration, the reference gas scale factor, the reference PPM, the slope of the line connecting the zero and reference points, and the date of the calibration. The zero and reference calibration points, and the calibration temperature are in ADC units and are used to compute the PPM. The set of points found during the Span and Zero Calibration is used in the computation of the estimated reference point during the Zero Calibration. The manufacturer when investigating the operation of the system uses the date of the calibration. Table 3 shows the layout of the data within the calibration table.

Table 3: Sensor Unit calibration table data layout.

Offset From Table Start	Table Element Name	Software Data Type
0	Slope	Microchip Floating-Point
4	Scale Factor	8-Bit Integer
5	PPM of Reference Gas	Microchip Floating-Point
9	Calibration Temperature	Microchip Floating-Point
13	ADC_ZERO_CAL during Span and Zero Calibration	16-Bit Integer
15	ADC_ZERO_CAL during Zero Calibration	16-Bit Integer
17	ADC_CAL during Span and Zero Calibration	16-Bit Integer
19	ADC_CAL during Zero Calibration	16-Bit Integer
21	Day of Calibration	8-Bit Integer
22	Month of Calibration	8-Bit Integer
23	Year (2-Digit) of Calibration	8-Bit Integer

4.22.3 Design of Limits Table

The limits table will contain the limits data, describing operation of features specific to the installation site and purpose. The limits table will contain the current altitude and the latching and acknowledge settings. The altitude represents the number of meters above sea level of the system. The latching and acknowledge setting are on or off values, used to control the alarm horns and external relays. Table 4 shows the layout of the data contained in the limits table.

Table 4: Data layout of the Sensor Unit limits table.

Offset From Table Start	Table Element Name	Software Data Type
0	Altitude	16-Bit Integer
2	Latching and Acknowledge Settings	8-Bit Integer

4.22.4 Design of Sensor Data Table

The sensor data table contains information specific to the Sensor Module and information for the long-term monitoring performed at the remote computer. The sensor data table contains parameters required for long-term monitoring, linear curve fits used in the calculation of the compensation multipliers, stability and validation parameters used during calibrations and sensor warm-up, manufacture date of the Sensor Module, maximum range of the sensor, and the operational and shelf life of the Sensor Module. The layout of the Sensor Data Table is presented in Table 5.

A single linear approximation to the background current and output temperature compensation curves did not provide the desired accuracy. The two linear approximations are fitted to the background current and two linear approximations are fitted to the output temperature compensation curves to provide greater accuracy. The current temperature determines which linear approximation is used for each curve. The background current curve gives 0-PPM below 20 C, the first approximation is used between 20 C and 30 C, and the second approximation is used above 30 C. The output temperature compensation curve uses the first approximation below 20 C and the second above 20 C.

The TWA-TLV and STEL values are used by the remote computer to perform the long-term monitoring. The remaining values are used by various routines to check input and calculated data. Additional data values are included providing flexibility for the addition of more diagnostics and to provide spare space allowing for future additions.

4.22.5 Design of History Data Tables

The system will store the previous two calibrations and configurations. Each time a calibration or a configuration is performed, the history data tables will be updated. The calibration or configuration previous 2 data table will be replaced with the calibration or configuration previous 1 data table. The calibration or configuration previous 1 data table will be replaced with the working calibration or configuration table. Finally, the working calibration or configuration data tables will be updated with the new information.

Table 5: Sensor Data Table layout.

Offset From Table Start	Table Element Name	Software Data Type
0	TWA-TLV Alarm Threshold	Microchip Floating-Point Number
4	STEL Alarm Threshold	Microchip Floating-Point Number
8	Background Current Curve 1 *a*	Microchip Floating-Point Number
12	Background Current Curve 1 *b*	Microchip Floating-Point Number
16	Background Current Curve 2 *a*	Microchip Floating-Point Number
20	Background Current Curve 2 *b*	Microchip Floating-Point Number
24	Output Temperature Compensation Curve 1 *a*	Microchip Floating-Point Number
28	Output Temperature Compensation Curve 1 *b*	Microchip Floating-Point Number
32	Output Temperature Compensation Curve 2 *a*	Microchip Floating-Point Number
36	Output Temperature Compensation Curve 2 *b*	Microchip Floating-Point Number
40	Output Altitude Compensation Curve 1 *a*	Microchip Floating-Point Number
44	Output Altitude Compensation Curve 1 *b*	Microchip Floating-Point Number
48	Maximum Allowed Slope Value	Microchip Floating-Point Number
52	Minimum Allowed Slope Value	Microchip Floating-Point Number
56	Slope Threshold	Microchip Floating-Point Number
60	Manufacture Date of Sensor Module (Day, Month, 2-Digit Year)	1 Byte for Day, Month, and Last Two Digits of Year
63	Factory Calibration Date of Sensor Module (Day, Month, 2-Digit Year)	1 Byte for Day, Month, and Last Two Digits of Year
66	Sensor Life (Months)	2-Byte Integer
68	Sensor Shelf Life (Months)	2-Byte Integer
70	Sensor Serial Number	12 Characters
82	Sensor Type	8 Digits
86	Minimum ADC_ZERO_CAL Value	2-Byte Integer
88	Maximum ADC_ZERO_CAL Value	2-Byte Integer
90	Maximum Stability Variation of Slope	1-Byte Integer
91	Minimum Peak Value	2-Byte Integer
93	Minimum Area Under the Curve	4-Byte Float
97	Magnitude Threshold	2-Byte Integer
99	Diagnostic Step Length	1-Byte Integer
100	Maximum Range of the Sensor	Microchip Floating-Point Number
104	Decimal Position	1-Byte Integer

5.0 IMPLEMENTATION AND TESTING

The software was developed according to the design presented in the previous chapter. The software was tested in small parts, as each part was developed. Larger routines were developed and tested after the small routines called by the larger routines were implemented and tested. This process was repeated until the software to control the Area Monitor and the Sensor Unit was implemented. After implementation of both units, the tests outlined in Section 3.2 were performed to verify the software against the specifications.

The software can be broken into three general sections, the software to control the transitions between states, the software to perform the computations, and the software to interface with the external hardware. Development of each of the above three categories of code progressed in parallel. This parallel development required three separate test platforms to verify each part before integrating the three parts to create a top-level routine.

A software simulation was developed to test the code controlling the transition between states. The simulator provided single step capability through the finite state machine, the ability to fire interrupt service routines, and stubs provided to mimic the I/O of the system. This simulation was used to debug and verify the correct operation of the finite state machine controlling the system.

The computations requiring verification are the calculation of the PPM, the calculations required to find the calibration point, and the calculations to determine when the warm-up tests have passed. Development and testing of the computations required development a software simulation. The calculation of the PPM was verified by simulating the code performing the PPM calculation in the MPLAB Sim software simulator. First, PPM calculation was performed in an Excel spreadsheet and then verified. Once verified, the input values used in the spreadsheet calculations were coded into the software and the resultant PPM value was obtained from the simulator after the PPM was calculated. The two values were compared to verify that both methods produced similar results. Software was developed to test the calibration and Sensor Warm-Up calculations because they both cycle through a loop once every second. In the case of the calibrations, the time must be kept to cause a fault if the point is not found within the specified time. The simulation halts once the duration expires. The simulation executed the loop as quickly as possible, but recorded the number of seconds required for the operation to complete and displayed the internal variables during each loop. Input simulating the sensor output during a calibration or warm-up period was used as input into the simulation. Exponential functions of the form $b*e^{at}$ were used to simulate the sensor output.

The device drivers to interface with the devices were tested by writing simple test programs to perform simple operations to verify that the device drivers are operating correctly. This testing was done using the final system and a PIC processor connected to a PC using RS-232.

The routines were gradually integrated to produce the final software. Full system functionality was then verified performing the tests described in Section 3.2. An In-Circuit-Emulator (ICE) was used to test and debug the software running on the PIC processor in the full system. The ICE allowed debugging of the system using the final system hardware. The ICE provided the ability to step through the code (single step, or run to a breakpoint) and allowed all internal registers and memory to be examined. Further system testing consisted of performing system operations in an order representative of the operational requirements of the system.

The output of the electrochemical sensor was simulated by applying a voltage source to the analog input to the ADC, in place of the sensor output. The voltage source provides precise control and monitoring of the sensor input to the system, allowing a thorough verification of the software. The calculation of PPM, the detection of an alarm condition, and transition to and from Alarm Mode were tested by altering the voltage input. The system responded to the changes in voltage by updating the PPM display and switching between Alarm Mode and Normal Operation Mode as the voltage was altered accordingly.

Zero Calibration was tested by performing a zero calibration at a known voltage input and then verifying that the PPM calculated after the zero calibration completed was close to 0-PPM. This test was performed using multiple input voltages.

The Span and Zero Calibration was tested by setting the voltage to a low value while the zero calibration was performed and then setting the voltage to a high value for the span calibration. After completing the calibration, verification consisted of ensuring that for input voltages around the low voltage the PPM was close to 0-PPM, and voltages around the high voltage, produced PPM was around the value specified for the reference PPM. The ICE provided the ability to verify that the calibration tables were updated and that the calibration history had been maintained as specified. The EEPROM was verified for both calibrations.

Power-Up and Sensor Warm-Up Modes were tested by applying power to the system and waiting for the system to enter Normal Operation Mode. Again, the voltage source was connected to the analog input of the ADC and was set below the magnitude threshold. The ICE was used to verify that the correct data had been downloaded and stored in the EEPROM.

Verification of Configuration consisted of ensuring that all data could be entered properly and that the Configuration data tables were updated properly. The ICE was used to inspect the value of the EEPROM on each processor. The configuration was performed and the execution was halted and the EEPROM was inspected to ensure that tables were properly updated. Next, the configuration was performed, changing all settings, without the ICE. After the configuration, the voltage input was altered to ensure the transition between the alarm states occurred at the new threshold levels.

A PC was used to interface with the Sensor Unit and Area Monitor using the RS-232 connection. The PC was used initially to debug the Area Monitor and Sensor Unit in isolation, using the PC to emulate the other part of the system. All operations from the remote computer were tested using the PROCOMM program for RS-232 communications on the PC. All functionality was verified using the PC before the Sensor Unit and Area Monitor were integrated.

The PC was connected to the Area Monitor to emulate the remote computer. Verification of Update Limits consisted of using the ICE to ensure that the data tables were properly updated. The data sent in response to the Download System Data was inspected to ensure that the data matched the test data in the system. The Request System State feature was testing in a similar fashion.

The software was developed using C and PIC assembly code. The PIC assembly code was used in parts of the driver functions. A number of development tools were used to develop the system. MPLAB IDE provided by Microchip was used as the development environment for the software. MPLAB IDE provided the means to program the PIC processors and debug code using the software simulator, MPSim and the ICE debugger, MPLAB ICE 2000. The CCS PCH compiler was used to compile the C code. The MPLAB ICE 2000 debugger toolset consists of the MPLAB ICE 2000 Emulator, the PCM18XD1 Processor Module, and the DVA16XP401

Device Adapter. MPSim is a software simulator that allows simulation of software without any external hardware. This simulator was used in the early stages of development to test the mathematical functions to verify the accuracy of their results. The PIC processors were programmed using the Pro Mate II Device Programmer, and the AC164012 Pro Mate II Socket Module. MPLAB IDE provided the interface to the Pro Mate II to program the compiled software into the PIC processor.

6.0 CONCLUSIONS

The software was designed, implemented, and tested based on the initial requirement to develop a system capable of monitoring hazardous gases with the ability to accept various types of sensors. Due to the system requirements, a distributed system was required. Software was designed, implemented, and tested to control the operation of the distributed system. The system developed was targeted at detection of EtO, but the ability to install multiple types of sensors was designed into the system. Each Sensor Module contains, in FLASH memory, the parameters needed for operation. Monitoring for gases other than EtO requires only the development of a sensor capable of monitoring the target gas, conditioning the sensor output to be within the system requirements, and placing the data for the target gas into the FLASH unit on the Sensor Module.

Communication protocols were developed to facilitate the transfer of data, commands, and messages between the system components. The distributed components that comprise the system must stay synchronized during transitions between states, preventing the incorrect interpretation of messages. The state machines controlling each component stay synchronized by preventing critical state transitions until the other component indicates it is ready, and through the use of a synchronization handshake with the other component. The communication protocol between the Area Monitor and the Sensor Unit incorporates a special synchronization exchange to prevent either component from advancing ahead of or falling behind the other component.

The system is designed to handle user input or react to messages from another component as quickly as possible. Providing the fastest response to user input, an alarm, or a fault is critical as it provides more time for users to correct a fault, or to evacuate after the detection of unsafe environmental conditions and to activate safeguards with minimal delay. Interrupt service routines developed for all time critical input devices, allow the system to respond to input immediately. This immediate reaction to input allows the system to respond immediately to user input, or to an alarm or fault message at any time.

The system developed provides for precise and improved monitoring of EtO in the environment. The system introduces new features and combines existing features to improve monitoring capabilities. The system compensates for altitude and temperature automatically when computing the PPM value. The estimation of the reference point during the zero calibration produces a SLOPE with improved accuracy as opposed to using the old reference point with the new zero point. The ability to locate the gas sensor portion up to 25-feet from the Area Monitor, allows the system to monitor locations difficult to access, and to display the PPM in an easily accessible area.

The built-in interface providing critical maintenance operations allows in-house technicians to perform routine maintenance operations on the system. Automated maintenance routines require minimal input and training for an in-house technician to learn how to perform maintenance, reducing the maintenance costs of the system. The connection to the remote computer allows the ability to monitor and control several monitoring systems from one central location.

BIBLIOGRAPHY

1. Mortimer, V. D. Jr. and Sharon L. Kercher. Control Technology for Ethylene Oxide Sterilization in Hospitals. U.S. Department of Health and Human Services. Washington: GPO, 1989.

2. United States. Department of Health and Human Services. Preventing Worker Injuries and Deaths from Explosions in Industrial Ethylene Oxide Sterilization Facilities. Washington: GPO, 2000.

3. United States. Department of Health and Human Services. Current Intelligence Bulletin 52. Washington: GPO, 1989.

4. Hilliker, D. J. "Employee Safety and Sterilant Gases: Are You Safe? How do You Know?" Retrieved May 13, 2004 from ChemDAQ Corporation web site: http://www.chemdaq.com/literature/are_you_safe_article.pdf.

5. United States. Department of Health and Human Services. Guidelines for Protecting the Safety and Health of Health Care Workers. Washington: GTO, 1988.

6. United States. Department of Labor. 1910.1000 Table Z-1 Limits for Air Contaminants. Washington: GPO, 1997.